$2.50 IN U.S.A
SPRING 1980 VOLUME 1 NUMBER 4

A McGRAW-HILL PUBLICATION

onComputing™

GUIDE TO PERSONAL COMPUTING

Spring 1980 onComputing VOLUME 1 NUMBER 4

MORE COLOR. MORE SOUND. MORE GRAPHICS CAPABILITIES.

ATARI 400

ATARI 800

Compare the built-in features of leading microcomputers with the Atari personal computers. And go ahead, compare apples and oranges. Their most expensive against our least expensive: the ATARI® 400™.

Start with graphics capabilities. The ATARI 400 offers 128 color variations. 16 colors in 8 luminance levels. Plus 29 keystroke graphics symbols and 8 graphics modes. All controlled from a full 57 key ASCII keyboard. With upper and lower case. And the system is FCC approved with a built-in RF modulator. That's just for openers.

Now, compare sound capabilities. Four separate sound channels and a built-in speaker. With the optional audio/digital recorder, you can add Atari's unique Talk & Teach™ Educational System cassettes.

Here's the clincher: Solid state (ROM) software. For home management, business and entertainment. Or just plug in an Atari 10K BASIC or Assembler language cartridge and the full power of the computer is in your hands.

Memory? 8K expandable to 16K. And that's just for the ATARI 400 at a suggested retail of only $549.99.

The ATARI® 800™ gives you all that and much more.

User-installable memory to 48K. A full-stroke keyboard. With a high-speed serial I/O port that allows you to add a whole family of smart peripherals. Including up to four individually accessible disk drives. And a high speed dot-matrix impact printer. And, the Atari Program Recorder is included with the 800 system. Suggested retail price for the ATARI 800 (including recorder) is $999.99.

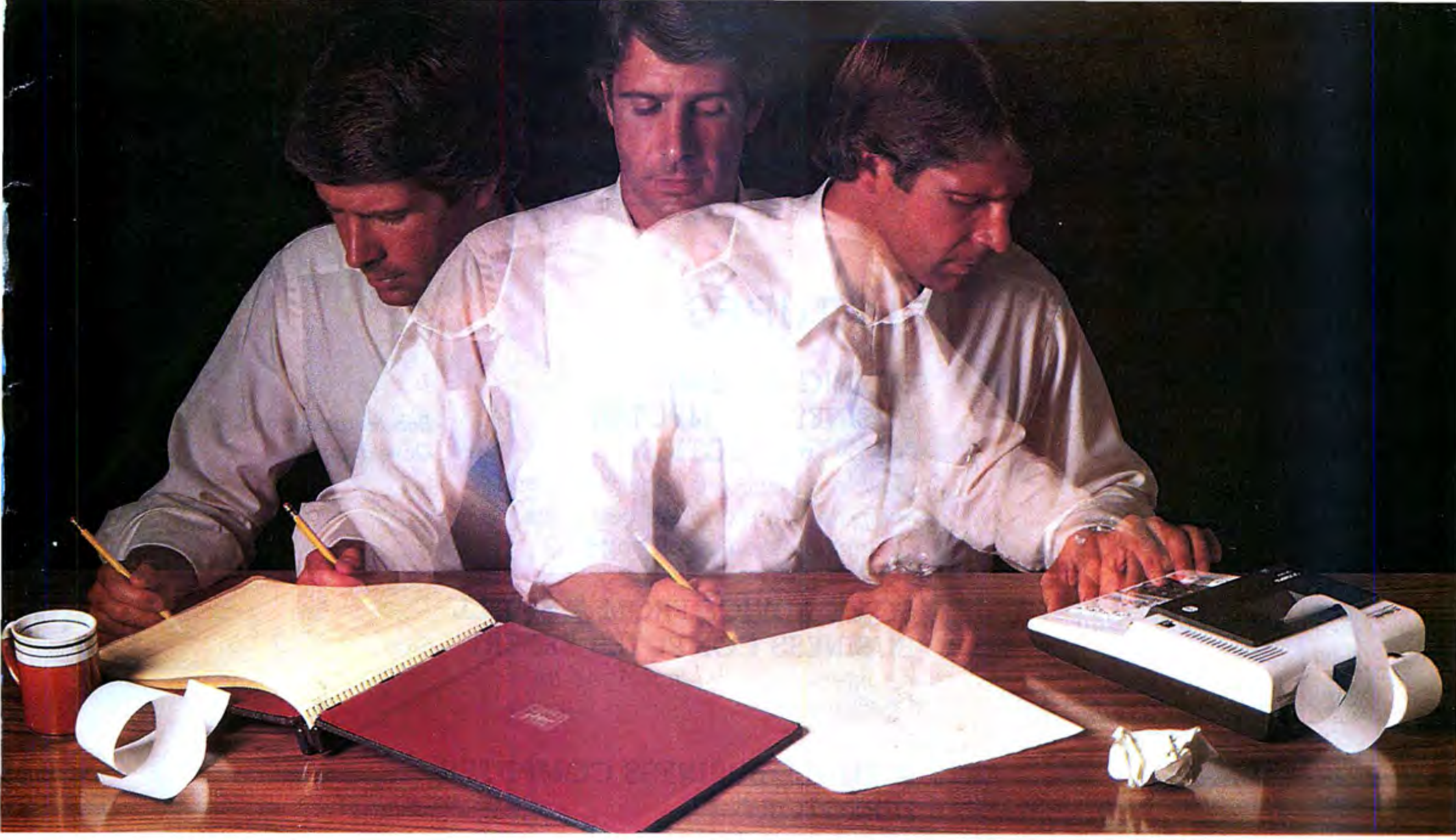
Make your own comparison wherever personal computers are sold. Or, send for a free chart that compares the built-in features of the ATARI 400 and 800 to other leading personal computers.

ATARI®

PERSONAL COMPUTER SYSTEMS

1265 Borregas Ave. Dept. C, Sunnyvale, California 94086. Call toll-free 800-538-8547 (in Calif. 800-672-1404) for the name of your nearest Atari retailer.

Circle 260 on inquiry card.



Solve your personal energy crisis. Let VisiCalc™ Software do the work.

With a calculator, pencil and paper you can spend hours planning, projecting, writing, estimating, calculating, revising, erasing and recalculating as you work toward a decision.

Or with the Personal Software™ VisiCalc program and your Apple* II you can explore many more options with a fraction of the time and effort you've spent before.

VisiCalc is a new breed of problem-solving software. Unlike prepackaged software that forces you into a computerized straight jacket, VisiCalc adapts itself to any numerical problem you have. You enter numbers, alphabetic titles and formulas on your keyboard. VisiCalc organizes and displays this information on the screen. You don't have to spend your time programming.

Your energy is better spent using the results than getting them.

Say you're a business manager and want to project your annual sales. Using the calculator, pencil and paper method, you'd lay out 12 months across a sheet and fill in lines and columns of figures on products, outlets, salespeople, etc. You'd calculate by hand the subtotals and summary figures. Then you'd start revising, erasing and recalculating. With VisiCalc, you simply fill in the same figures on an electronic "sheet of paper" and let the computer do the work.

Once your first projection is complete, you're ready to use VisiCalc's unique, powerful recalculation feature. It lets you ask "What if?," examining new options and planning for contingencies. "What if" sales drop 20 percent in March? Just type in the sales figure. VisiCalc instantly updates all other figures affected by March sales.

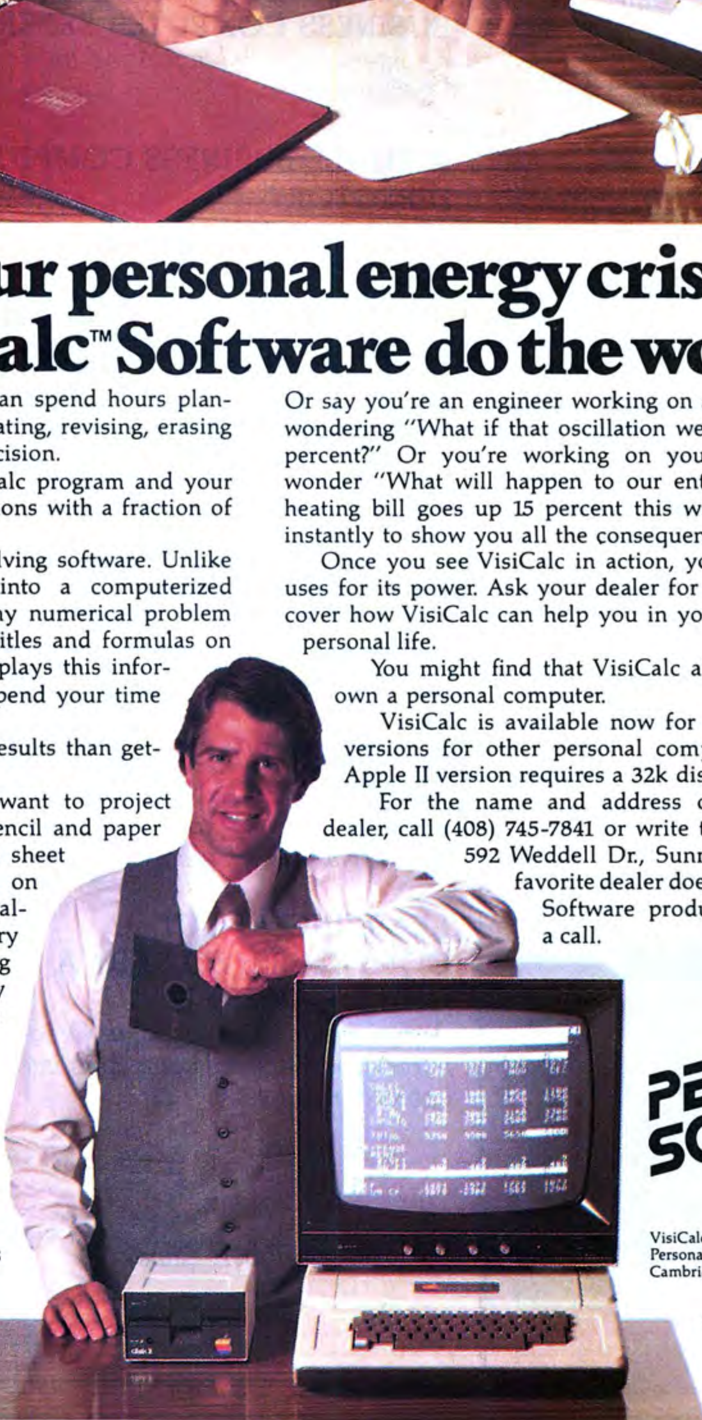
Or say you're an engineer working on a design problem and are wondering "What if that oscillation were damped by another 10 percent?" Or you're working on your family's expenses and wonder "What will happen to our entertainment budget if the heating bill goes up 15 percent this winter?" VisiCalc responds instantly to show you all the consequences of any change.

Once you see VisiCalc in action, you'll think of many more uses for its power. Ask your dealer for a demonstration and discover how VisiCalc can help you in your professional work and personal life.

You might find that VisiCalc alone is reason enough to own a personal computer.

VisiCalc is available now for Apple II computers with versions for other personal computers coming soon. The Apple II version requires a 32k disk system.

For the name and address of your nearest VisiCalc dealer, call (408) 745-7841 or write to Personal Software, Inc., 592 Weddell Dr., Sunnyvale, CA 94086. If your favorite dealer doesn't already carry Personal Software products, ask him to give us a call.



PERSONAL SOFTWARE

VisiCalc was developed exclusively for Personal Software by Software Arts, Inc., Cambridge, Mass.

Circle 10 on inquiry card.

TM—VisiCalc is a trademark of Personal Software, Inc.

*Apple is a registered trademark of Apple Computer, Inc.

onComputing

Volume 1
Number 4
Spring 1980

FEATURES

Cover by Robert Tinney

PRODUCT REVIEWS

DEPARTMENTS

Officers of McGraw-Hill Publications Company: Paul F. McPherson, President; Executive Vice Presidents: James E. Boddorf, Gene W. Simpson; Group Vice President: Daniel A. McMillan; Senior Vice President-Editorial: Ralph R. Schulz; Vice Presidents: Kemp Anderson, Business Systems Development; Stephen C. Croft, Manufacturing; Robert B. Doll, Circulation; James E. Hackett, Controller; William H. Hammond, Communications; Eric B. Herr, Planning and Development; John W. Patten, Sales; Edward E. Schirmer, International.

Officers of the Corporation: Harold W. McGraw Jr., President, Chief Executive Officer and Chairman of the Board; Robert F. Landes, Senior Vice President and Secretary; Ralph J. Webb, Treasurer.

onComputing is published four times a year by onComputing Inc, 70 Main St, Peterborough NH 03458, a wholly-owned subsidiary of McGraw-Hill, Inc. Address all mail except subscriptions to above address: phone (603) 924-9281. Address all subscriptions, change of address, USPS Form 3579, and fulfillment questions to onComputing, POB 307, Martinsville NJ 08836. Second class postage paid at Peterborough NH 03458 and at additional offices. (ISSN 0194-8075) Subscriptions are $8.50 for one year in USA and its possessions. In Canada and Mexico $10 for one year. Other foreign countries $12 for one year surface delivery. Single copy price is $2.50 in the USA and its possessions, $2.90 in Canada and Mexico, $4 in Europe, and $4.50 elsewhere. Foreign subscriptions and sales should be remitted in United States funds drawn on a US bank. Printed in the United States of America.

Address all editorial correspondence to the editor at 70 Main St, Peterborough NH 03458. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of onComputing.

Subscription WATS Line: (800) 258-5485

Editor's Message

Small Business Computers and Other Matters

One of our major tasks here at onComputing is to keep track of new personal computer trends in home applications, education, computer toys, the professional field — in short, a whole world of applications.

In this issue we explore one of the fastest growing areas in the field: small business computers. Suddenly, a flood of new hardware and software has descended on the marketplace, and everyone seems to be offering Accounts Receivable packages.

Most of the small business systems that sell for under $10,000 are really designed for the "very small business" market (ie: those companies having fewer than ten employees each). There are 2.3 million such companies in the United States alone. A surprising number of them are discovering the advantages of computers in their daily operations. McGraw-Hill's Research Department has calculated that over 100,000 personal computers were sold last year for business applications alone.

Actually, the term "business computer" is somewhat of a misnomer — a computer is a computer is a computer. The real differences between small business computers and other small computers lie in software, the sophistication of the peripherals used (eg: a letter-quality printer may be needed for printing correspondence), and the amount of data storage needed (businesses often need to store huge quantities of data).

We've tried to sort out some of the clutter and confusion with a special article by Bob Magruder and Dave Barstow of Advanced Systems Group in Madison, Connecticut, called "Buying a Small Business Computer: An Introduction." Bob and Dave are consultants to the business community and have an abundance of practical information to share with onComputing readers. We offer, too, some advice from MicroAge's Jeff McKeever, a small business computer expert.

Also in this issue is a comprehensive directory of small business computers and software put together by Charles Freiberg — a superhuman task in the face of a constant barrage of new products being announced nearly every day.

* * *

"Profile: The Small Town Doctor and the Personal Computer" is a record of my unforgettable trip to Poynette, Wisconsin, last year. Dr McNeill, head of the Poynette Family Clinic, uses a personal computer in his general practice for much more than just billing patients. His entire approach to computers is unconventional, and his ideas are beginning to have an effect on the medical community. I was struck by his fierce individualism and his dedication to both of his crafts: medicine and computer science.

In this issue we also evaluate OSI's new color computer, take a look at computer mail in Alaska, introduce you to programming languages, talk about personal computers in Vietnam, explore computers in education in an Oregon school, and rate Radio Shack's new TRS-80 Model II computer.

* * *

We appreciate the feedback — both positive and negative — that we have received from you, our readers, over the past few months. Creating a new magazine is a challenging task, and it is gratifying to know that you are alert to and interested in the growing world of personal computers. We plan to continue giving you the latest and most accurate information we can.

Chris Morgan

Chris Morgan
Editor-in-Chief

Letters

onComputing welcomes correspondence from readers. Letters should be addressed to onComputing, 70 Main Street, Peterborough NH 03458.

Talcott Mountain

Dear Editor:

Thank you for an informative, enjoyable, and down-to-earth computer magazine. onComputing does much to alleviate the unnecessary mystique that so typically surrounds the world of computers.

Having visited the Talcott Mountain Science Center, I was particularly pleased with Daniel Barstow's recent article (Winter 1979 onComputing, page 34) describing their exemplary program. I find it refreshing to see computer applications in the classroom extended beyond the "learning to program as an end in itself" approach.

Hats off to a fine program and a delightful article!

Beth Stoddard
Springfield MA

Point-of-Sale and Other Matters

Dear Editor:

I would first like to compliment you and all of your people on putting together a first-class magazine. It is very refreshing to be able to read a magazine and understand what the author of the article is saying without having to refer to your 6500 programming manual.

Second, your new product announcement on page 89 in the Winter 1979 onComputing about the High Technology Point-of-Sale software package is incorrect. The software belongs to Apple Computer Inc. It is called "The Cashier" and retails for $250.

Third, I would like to solve the reset problem of Mr Jim Fleming (see the Winter 1979 onComputing Letters, page 80) and thousands of other Apple users. We manufacture and sell a reset remover. It is a simple device that allows you to enable or disable the keyboard reset at the flick of a switch. We retail these for $29.95. Any Apple owner can contact us direct or go to his or her local Apple dealer and get one. Our address is Computer Solutions, 5135 Fredericksburg Rd, San Antonio TX 78229, 1-512-349-8851.

John Gaines
San Antonio TX

Praise for onComputing

Dear Editor:

Congratulations! Your first issue, which I have just finished reading, was the most helpful of all the recent publications I have come across for people like me who are in the "just starting to think about whether I should get a computer" stage.

James A Levine
Wellesley MA

Speedy Chess

Dear Editor:

onComputing fills an important gap in the spectrum of microcomputing magazines by providing information for the growing number of newcomers to personal computing. However, since the neophyte is more likely to accept what he reads as gospel, you should strive harder than the other magazines to maintain high standards of accuracy.

This criticism is prompted by an error in John Martellaro's otherwise excellent article on computer chess in the Winter 1979 issue. Mr Martellaro ascribes the slowness of *Sargon* on the Apple II to the fact that the Apple's 6502 microprocessor runs at only 1 MHz, compared with the 4 MHz of the Z80 in the Jupiter computer that *Sargon* was originally written for. He also seems surprised that "...even though the TRS-80 runs at 1.77 MHz..." its version of *Microchess* is not as strong as the Apple version.

While he does give other reasons for *Sargon's* slowness, Mr Martellaro is propagating a misconception here. Not that he shouldn't compare the speeds of the various programs, but it is misleading to imply that the speed differences are related to the clock speeds of the microprocessors, except where the same microprocessor is being run at different speeds. The simple fact is that *you cannot draw any conclusions about the relative performance of different microprocessors by comparing their clock speeds.* For example, the Z80 is generally considered to be more powerful than the 6502, but the Apple's 6502 running at 1.023 MHz is actually faster than the TRS-80's Z80 running at 1.776 MHz.

The only way to compare computer speeds is to run equivalent programs on them and see how they perform, but if the programs aren't equivalent, you can't draw any conclusions about the computers themselves. In the case of *Sargon*, it would be interesting to find out just how bad the 6502 version running on the Apple II is, compared with the Z80 version running on the Jupiter Wave-Mate; Mr Martellaro doesn't say. If the 6502 version was completely rewritten, it might not be much slower than the Z80 one. On the other hand, if the 6502 program was converted directly from the Z80 code it is bound to be significantly slower, since the programming techniques for these two microprocessors are radically different.

Allen Watson III
Sunnyvale CA

Mr Martellaro replies:

Contrary to my statement in "Sargon vs Microchess," the clock speed of a microprocessor is not, in general, a measure of the speed of computation because the microprocessor architectures vary so widely. A good example of this, which I should have thought of, is how fast Microsoft BASIC executes on the Apple II.

Regarding the application to chess, see for example Personal Computing, *September 1979, page 60 which has the comment: "Dan (Spracklen) thinks the 6502 has a faster* effective *clock speed for chess applications than the Z80." Emphasis mine.*

This means that for the type of operations one has to do in a chess program, the result may be achieved faster with the particular architecture of the 6502 despite its lower absolute clock speed of 1 MHz.

My regrets for not doing my homework more thoroughly.

TBS-80 GENERAL ACCOUNTING SYSTEMS. ONE STEP BEYOND.

Your TRS-80™ microcomputer is not a toy. These **TBS-80 general accounting systems** aptly demonstrate the power of your computer.

ANALYSIS PAD by Del Jones is the epitome of first-class programming in business applications. Requiring 48K, and one disk with a printer recommended, this columnar calculator gives the user tremendous flexibility in data entry, enabling the user to create 30 or more columns and rows. Enter your own column and row labels. Enter your data by row or column or directly onto screen display via edit mode. Move, swap, delete, and add rows or columns. Create new pads by stripping relevant data from old files. You never have to key in data twice. But, more important than the powerful data manipulation provided, add, subtract, multiply and divide one column by another and put results in another column. Perform up to six calculations on one column and even define one column to be a constant. The calculation routine you create can be saved and reused. Print out the entire pad in four column segments to line or serial printer. **ANALYSIS PAD** was originally advertised for 32K tape at $32.50. Since then, it has been totally rewritten and expanded to its present 48K disk only form and sells for $49.50. It is easily worth twice as much. You have to see it to believe it.

CHECK REGISTER ACCOUNTING SYSTEM, adapted for the TRS-80 by Dale Kubler and originally written by O.E. Dial, is the most comprehensive check-balancing program written. Requiring 32K, two disks and printer, this program does much more than just balance and reconcile your checkbook. It enables you to define up to 60 account names and will generate monthly summaries of all accounts with monthly and year-to-date totals. Single-entry input allows the user to disperse one transaction over several accounts and to make a 64-character note on each transaction. Checks can be printed out after data has been entered. Aside from the Statement of Accounts, **CRAS** also generates the following reports: Check Register for any Month, Notes to Check Register, Income/Expense Distribution, Statement of Selected Accounts, Bank Reconcile Statement and Suspense File. The Suspense file is an extra feature where you can make notes to yourself for any month in the year. **CRAS** will make both you and your account happy and it sells for $49.50.

CHECKBOOK II by Alan Meyers is the finest program of its kind yet published. With superb graphic screen displays, it does everything necessary to keep your checkbook balanced. Data is input directly into a five-column screen display with a field for alpha or numeric codes. Editing is done easily for changes in any or all columns. **CHECKBOOK II** will accurately balance and reconcile your checkbook, handling balances up to $1,000,000. Your balance brought forward is always in memory. Outstanding checks are listed and easily saved. You can also search for an entry by any field except amount, and all checks with matching entries will be displayed and totaled. A numeric sort routine is included. Screen prints can be made to a line printer from almost any point in the program. In addition, the 32-48K version can write files to disk. This, and the 16K version, are included on the same tape. For $18.50, **CHECKBOOK II** is the bottom line in personal checkbook programs. A disk version of this program is available for $28.50.

BUDGET II (not yet released) by Alan Meyers, takes off where CHECKBOOK II ends. Written exclusively for either disk or tape based computers, this program enables the user to set up 20 account names with four character codes for each, that correspond to the codes used in Checkbook II. Each account can be tagged income or expense and whether it is fixed or not. Set your monthly budget and balance it. Disperse your cash account over the other accounts. Checkbook II data is brought in and summarized by account and compared to amount budgeted. Year-to-date totals are included in monthly summary. Year Summary gives monthly and year totals for each account at a glance. Forecast feature enables user to enter rate of inflation and income increase to see financial standing after 12 months. Review enables user to go back and look at months previously summarized. Flashy graphics and much more. For 16K and 32K tape, **BUDGET II** sells for $24.50. For 32K up disk, $34.50. If you have CHECKBOOK II, you will want this program.

TBS has other incredible software for Tandy's microcomputer. Intent on making it a powerful tool, we have **large scale business accounting systems, data processing systems, system utilities,** and the **Library 100.** We have the only **DISK HEAD CLEANER** (for APPLE too!) and **GRAN MASTER DISKETTES,** the best on the market.

TBS is **YOUR COMPANY,** and we build systems, not just software. The above products are available now, nationwide. Visit your local Computer Dealer or Associate Radio Shack Store and demand the best, demand TBS. For more information, contact us through the numbers below.

™ TRS-80 is a trademark of the Tandy Corporation.

TBS™ **THE BOTTOM SHELF, INC.**
(404)939-6031 • P.O. Box 49014 • Atlanta, GA 30359

Circle 30 on inquiry card.

Buying a Small Business Computer: An Introduction

by Bob Magruder and Dave Barstow Advanced Systems Group

Suddenly, small business computers are everywhere, and buyers are simultaneously elated and confused over the wealth of new small business hardware and software. What should you buy? How much should you spend? $500? $12,000?

Anyone thinking of buying a small business computer system is faced with several problems. For one thing, *everyone* seems to have his or her own idea about which system is best. For another, recommendations may be given in what seems to be a strange foreign language.

When the buyer approaches the computer dealer, he or she will find

photo by James Caccavo

Example of a small business computer system. Shown are a North Star Horizon computer with two 5-inch floppy-disk drives, an NEC Spinwriter 5510 printer, and a Soroc IQ 120 video terminal. The system costs $8595, including a word processing software package. A hardware and software package such as this would suit the needs of a company generating up to $500,000 a year in business. (Picture taken at the MicroAge store in Scottsdale, Arizona.)

With companies like Radio Shack and Apple aiming their marketing guns at the small business user, the small business computer has suddenly come into its own.

a plethora of computer riches ranging in price from $500 to $12,000, and even higher. Included are kits, programs, boards, and manuals, all of which tend to confuse the issue even more.

Why Use Computers?

The key question is: What do you really need a computer for? The answers are different for computer hobbyists, personal computer users, and business computer users. They depend to a great degree on the experience of the potential buyer.

Let's begin with some definitions. The following definitions are not used throughout the industry, and may not be acceptable in technical circles, but they are nonetheless useful:

- Computer Hobbyist - A person who delights in taking electronic parts along with a few instructions and creating a computer or other electronic device. The hobbyist is often an experienced computer person in his or her own right. Computer hobbyists create many new and imaginative devices, and many would rather build than buy. (The availability of cheap "appliance" computers is changing this trend somewhat.)
- Personal Computer User - A person using a computer for home entertainment, games, education, or computer assisted instruction. The personal user may have some of his or her business records on the machine, but the machine is not indispensable.
- Business Computer User - A person who uses a computer in order to operate his or her business. Loss of the machine for more than a few days can mean a large financial loss to the owner.

When a computer hobbyist's machine breaks down, he or she can usually fix it, or at least diagnose the ailment. Personal computer users and small business system users, on the other hand, are often not able to do their own repair work. When a personal computer user's machine is "down," it is definitely an inconvenience. But for the small business user, it can be a catastrophe! ("Down time" refers to the time when a computer is out of service.)

One important consideration in purchasing a computer of any size is dependability. Another criterion is its intended use; a third is the service available on the various components of the machine.

There are three different types of computers: *mainframes, minicomputers,* and *microcomputers.* These categories overlap, but some arbitrary definitions can be established:

- Mainframe - A large computer, usually housed in a special temperature-controlled room. Mainframe computers are the traditional computers familiar to most people. In their various forms, they have been around for several decades, and generally combine high performance with high price (from $250,000 up to several million dollars). Mainframe computers are usually run by computer professionals. The two major advantages of mainframe computers are speed and capacity. IBM's machines typify this category.
- Minicomputer - This intermediate computer has become more and more attractive to business users over the past few years. Priced roughly between $20,000 and $250,000, minicomputers approach the speed and capacity of mainframes, but cost much less. Digital Equipment Corp and Data General Corp are two major manufacturers of minicomputers.
- Microcomputer - The smallest and least expensive type of computer, the microcomputer is the child of the electronic revolution of the early 1970s. At that time it became possible to put an entire computer in a handful of integrated circuits. These *small systems* offer a surprising number of features previously found only in minicomputers and mainframe computers. Microcomputers for small businesses typically cost from $2000 to $20,000.

The important differences be-

Bob Magruder has been associated with computers and computer user's problems since 1959. Bob presently runs two firms in the microcomputing field: A-V Systems Group and Advanced Systems Group, Madison CT. AVS provides consulting services in the uses of microcomputers to support educational processes, and ASG provides consulting services for commercial and personal users.

Dave Barstow has been involved with computers as a designer, as well as a user, primarily in the fields of communications and control. Dave is presently Senior Associate, Advanced Systems Group, Madison CT.

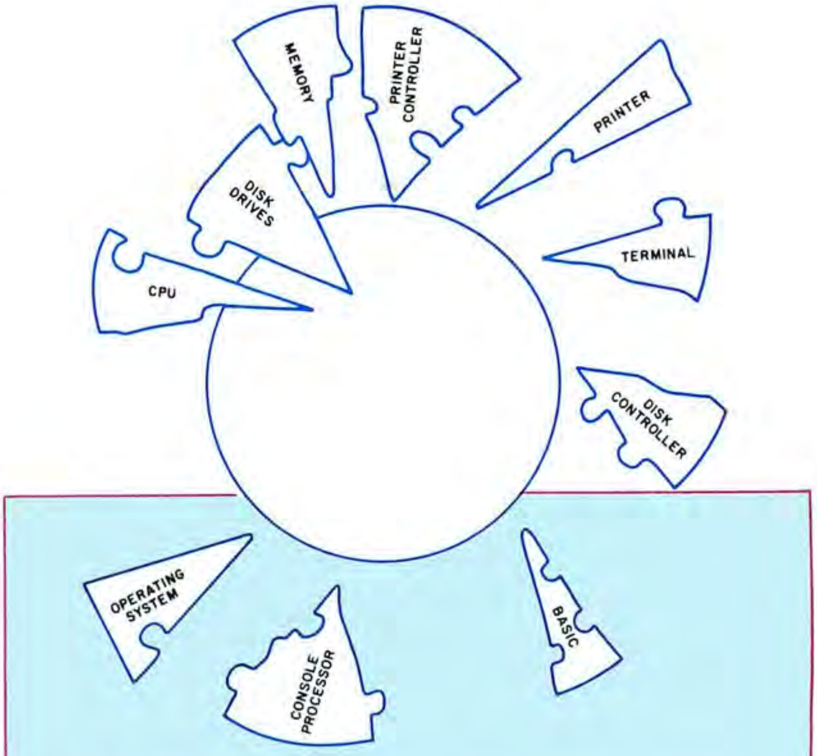


Figure 1: Pieces of the puzzle. Many of today's small business people are assembling their own computer systems to ease the paperwork and other drudgery of their small businesses. Unlike the buyer of large computers, though, small system buyers often have to "do it themselves."

Graphs by Bill Morrello

tween mainframes, minicomputers, and microcomputers are cost and speed. Cost is a factor in every installation. Speed is usually a factor in business installations, in which a specific amount of work has to be done in a specific time. If a user has to enter 10,000 transactions daily from twenty to thirty terminals, a mainframe is really the only choice available. The user needs the speed and flexibility afforded only by a large, fast machine. If the user needs to enter 1000 transactions from four or five terminals, then a minicomputer would do the job, but a microcomputer would not. If the user needs to enter up to fifty transactions a day from one or two terminals, then a microcomputer will do the job.

The final difference between the three types of computers is the *human engineering* factor. The mainframe is a big, complicated, and rather intimidating machine in human terms. It is usually isolated from the user by a staff of computer professionals who run and maintain it. Unless users are willing to write their own programs, they must adapt their business practices to programs already running on the mainframe computer. Even while on a timesharing terminal and working directly with the machine, the user is always under the control of either the operator or the *executive program* in the machine. The executive program schedules work, allocates memory, and allocates the devices (disk drives, tape drives, printers, etc) to users.

Working with a minicomputer is much the same. An executive program controls the use of the machine. A great deal more flexibility in program design is available with a minicomputer, though, primarily because it is usually installed to service a single user. The user has quite a bit to say about program design and how programs handle data. Even so, the design, complexity, and cost usually inhibit the user from direct participation with the system. Changes are difficult and expensive to make. The minicomputer serving a business today can be quickly outdated, both because of technology changes and because changes in business practices have not been incorporated into program design.

Working with a microcomputer is different. The microcomputer is slower and smaller, making it seem more "friendly." Microcomputers rarely have an executive program to manage users, so the user has to tell the machine what to do. Microcomputer programs—at least the better ones—allow the user to modify headings in reports and data collected to reflect changes in the situations or environments in which they are used. Most important, with a microcomputer you are part of the system. You make it go or stop. You decide what is to be processed, and in what order.

What Will You Do With It?

The last set of criteria is perhaps the most important. To be able to evaluate any computer for home or business use, the user must be able to tell the computer professional or retailer exactly what the machine has to do.

Is the machine supposed to keep payroll accounts, update mailing lists, and prepare summary reports? The more detailed the user's "shopping list" is, the more likely he or she is to get the right machine, of proper size, and with the correct storage media. For example, if graphs and maps are to be plotted, high-resolution graphics might help to interpret the results. Color may or may not be required, depending on the complexity of the plot or use of the map.

Potential business users should

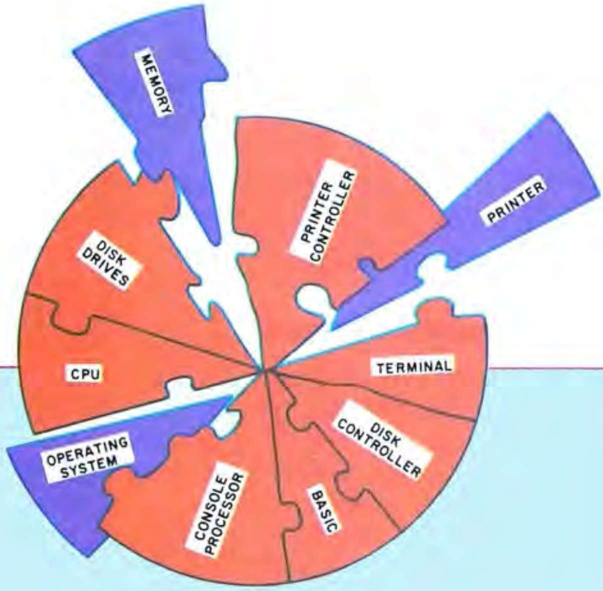


Figure 1a: Even after the pieces of the puzzle have been purchased, they might not fit together smoothly. There could be hardware problems, software problems, or both. Depending on the user's technical background (or lack of it), he or she should reexamine the system, or talk to a consultant or computer store about the problem.

know how many transactions will be posted on the machine daily, how many checks will be written, what kinds of reports are desired, and so on. If a word-processing station is needed, what kinds of data will be stored and how will it be used? These are not easy questions—they require that a business examine its procedures and determine precisely what happens every business day.

On one hand we have some computer criteria; on the other, some use criteria. If we compare them, we should get guidelines for selecting a computer system.

If you are a potential personal computer user, you might consider an "appliance" computer such as the TRS-80, PET, Apple, Atari, and so on. These machines are widely available and therefore highly serviceable. They are also relatively inexpensive, depending on the number of options chosen. For the most part, they are competitive in price, and the choice can be made based solely on intended use. A large number of programs are available from the manufacturers and from outside sources. Scores of computer clubs and special interest groups are oriented to the problems relating to the use of these machines, especially by the first-time user. The machines and their instruction books are oriented to the first-time user. We strongly suggest that a minimum of 16 K bytes of programmable memory be installed to enable you to use the complete BASIC programming languages available on these machines.

Small business users need to look first at their own operations to find out what the computer is supposed to do. They should also look at their business growth. The decision to buy a computer for a business should really be weighed against the need for additional personnel. A computer doesn't take coffee breaks or go to lunch, but it is important to consider the complete cost of adding one to the staff. A computer won't work without programs (ie: *software*). Specific software for particular business situations is sometimes not generally available. Generalized software is available for larger machines. It may not meet the needs of most business applications, and will probably have to be changed to meet specific situations. Such changes are part of the cost of a computer for the small business.

Finally, the buyer should look very carefully at the computer operator, who should be both knowledgeable about the business and enthusiastic about learning how to operate the machine. The operator is also a part of the cost of operating the machine, although all of the operator's time should not be charged to the machine unless the operation will be a full-time job.

With this information in hand, the new business user can look at machines, software, and people as a unit (the "system") and provide the computer consultant or dealer with an idea of what the demands on his or her machine are going to be. We recommend that new business users seek competent help in selecting a system. Competent help can be found by asking a professional or retailer for references before committing funds to a machine. How many installations has the dealer managed? Who among his customers have had problems similar to yours? Most dealers and consultants will gladly provide references. Those that won't usually have one of two reasons not to. Either they had a similar job that went wrong, or they have never performed such a job in the first place and want "on-the-job training." Training "experts" is very expensive.

Business operations should con-

sider a heavy-duty machine. Two good examples are Cromemco and Dynabyte. The heavy-duty machine often uses "overspecified" components to decrease service problems and accommodate system expansion. System expansion is an important consideration. The system should grow as the business and its needs grow. If growth can take place, the potential for obsolescence because of changes in the business is greatly reduced. The minimum memory size we recommend for a business operation on one terminal is 48 K bytes of programmable memory. If a second user is to be accommodated in the next few years, the machine should be expandable to 64 K bytes for each user. The Cromemco is a twenty-two-slot machine (meaning that it can accept a total of twenty-two printed-circuit boards) which can accept up to 516 K bytes in internal memory. The Dynabyte can use up to 216 K bytes in internal memory. Both machines are highly expandable.

What About Peripherals?

Manufacturers offer a variety of *peripherals* for the small business computer. A peripheral is an auxiliary device that attaches to a computer. Printers, floppy-disk drives, cassette recorders, and video terminals are examples of peripherals.

The tape recorders used in microcomputing are, primarily, inexpensive machines designed for the reproduction of voice. The better machines for data reproduction are usually the lower- to medium-priced portable tape recorders. More expensive tape decks do not usually work as well for data reproduction because they are designed to reproduce music and are designed with wide frequency response, which can actually be detrimental to data reproduction. Data signals, on the other hand, are identified by rapid changes in frequency. Usually then, the more expensive the tape recorder, the more it degrades the signal that the computer needs. Many appliance computers are supplied with a tape recorder or cassette deck to go with the machine (ie: Radio Shack TRS-80, Atari). Some, like the Apple computer, are not supplied with a tape drive. Cook Laboratories in Norwalk Connecticut, who record computer programs commercially, have reported excellent results with a $49.95 Sony portable tape recorder.

System expansion is important. The system should grow as the business and its needs grow.

Disk storage adds a new dimension to small business computers. Programs or data can be read into the computer much more quickly from disk than from tape. Data can be read from and written to disks almost at will, or at least it seems so. New freedom and new computing power result from the addition of disk drives to the system. Disk drives enable the user to access data files in any order. These files are called *random access files*.

Random-access storage allows the computer to decide where to put a file of information on a disk. The computer writes the file in the closest available blank space. To do this it must be able to tell which file is which and where files start and end. Random-access files, in most computer systems, are of fixed *record* length and contain a fixed number of *fields* per record. The computer is told by the program how long the record is and where the fields are in the record. The machine keeps a directory of the files on the disk and where they start and end. The computer finds a record in a file by going to the directory, and then quickly comparing a record in the file with the variable specified by the program. Random-access files allow faster data retrieval, but only if the files are properly organized for random access, *and then* only if the programs being used on the computer allow for the creation, writing, and reading of random-access files.

Another trade-off to consider is the speed of random-access files versus the requirement for file organization needed to find the files on a disk.

Buying disk-file storage used to be relatively simple. Once the size of the files was fixed, the size of the single-density disks needed to use the files was determined. (Two sizes exist: 5-inch floppies and 8-inch floppies.) The 5-inch floppy holds about 150,000 bytes and the 8-inch floppies about 350,000 bytes. (A bit is a single binary character, either 0 or 1. A byte is an 8-bit word in most microcomputers.)

Next, floppy-disk drive manufacturers announced *dual-density*, multitrack recording, and *dual-sided* disk drives. In theory, the manufacturers had increased the storage on any disk by a factor of eight. In practice, though, efficiencies are much lower because of the way disk records work. We are presently running a dual-track, dual-density, single-sided system which allows 315,000 bytes of storage on each 5-inch disk. Most 8-inch systems run between 1,100,000 and 1,300,000 bytes per dual-drive system. More reliable dual-sided systems are beginning to appear.

The big news on the disk drive horizon is the *hard* disk. Hard disks may be either permanently mounted on the drive and sealed, or a part of the disk-pack can be removable. Hard disks are faster, bigger, and more expensive than floppy disks. They are sized by megabytes (M bytes) which are millions of bytes of storage. Usual sizes are 5, 10, and 20 M bytes. Presently, the price of hard-disk storage varies between $500 and $1000 per M byte. However, this price will drop drastically as more units are produced. A 20 M byte hard-disk drive is commonly housed in a large floor-mounted cabinet. Most of those presently in production require temperature and humidity control within narrow limitations. All of our correspondence, legal contracts, forms,

and articles for our own business are on one disk which has been in commercial use for six months. It is just beginning to fill up. We will certainly run out of file numbers on the file before running out of room.

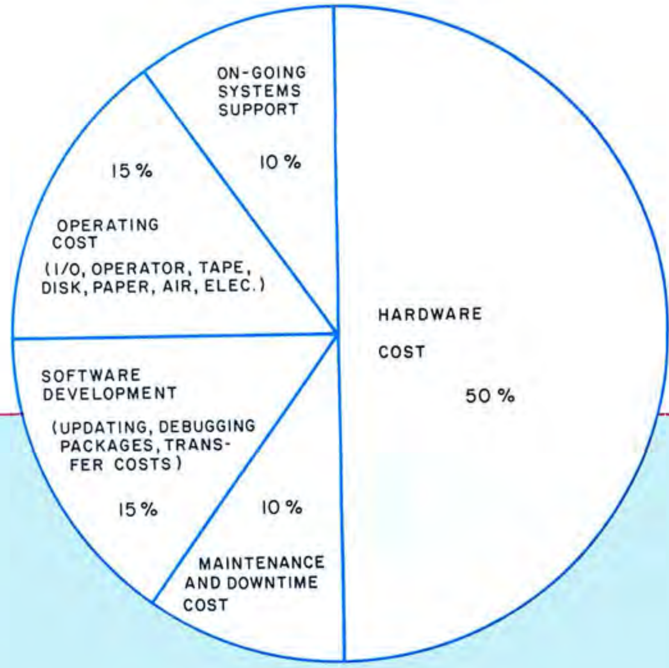


Figure 2: Beware of hidden costs. On the average, the cost of the hardware in a small business computer installation is only half of the total cost of getting the system running and keeping it running.

There are two types of printers in common use with computers. One prints full characters that look like they came from a typewriter or print shop. This is called a full-character printer. The other prints characters that look like they came from a computer, and are created by printing dots in the shape of letters. This type of printer is called a dot-matrix printer. *[See "A Printer Primer" in the Fall 1979 onComputing. . . .ed]*

The IBM Selectric typewriter conversion produced by I/O Unlimited and the Anderson-Jacobson printer, which uses a rebuilt Selectric, are examples of full-character printers. Both can be obtained in versions to work with almost any computer. These printers run at the maximum speed of the Selectric typewriter, which is limited by its mechanical design to about 14.9 characters per second (cps). The rebuilt machine costs about $900. The new machine costs about $1900. Both are "pinch-feed" machines. They do not require paper with holes up the edges as do "pin-feed" and "tractor-feed" machines.

To go faster with a full-character printer costs more money. The Xerox/Diablo printer costs about $2800 and will run at about 40 cps. This printer is called a "Daisy Wheel" printer because the print element is spread out and flat like a daisy. One of the fastest full-character impact printers available today is the NEC Spinwriter, which operates at 55 cps in both directions, and also has an "intelligent" carriage return. This feature keeps the machine from returning beyond the beginning of the next line. The effective printing rate of the Spinwriter is about 100 cps because of these features. Costs vary, but the basic machine is about $3000. Both the Diablo and the Spinwriter are available in "pinch-feed" or "tractor-feed" versions. These machines also may have the tractor removed so that they will print on standard letterhead paper. Full-character printers are primarily used for word-processing applications in businesses, such as accounting, consulting engineering, and law, where the appearance of the printed product is especially important. The potential printer user should watch the market very carefully, because new models are being announced nearly every week.

Dot-matrix printers are considerably faster than full-character printers. The Integral Data Systems IDS-125, for example, costs $750, and prints 10 characters per inch at a rate of 100 cps. Like most dot-matrix printers, this machine prints a variety of print sizes. The smaller the print size, the faster the printer prints. Several of the Centronics printers run at about 120 cps with 10 characters per inch at a cost of about $1200. The bidirectional Centronics printers run at more than twice this speed because the carriage does not have to return between each line of type. Effective printing rates approach 250 cps, but cost also doubles to around $2400. The fastest a dot-matrix character printer can print is about 255 cps. At that speed, the printer is running almost at line printer speeds.

Most of the line printers in use today use either band or drum type print mechanisms. Some line printers are dot-matrix printers that effectively have separate print heads

photo by James Caccavo

This small business system features a hard disk for storing data (CDC Phoenix hard disk with 3 megabytes of storage), a North Star computer, NEC Spinwriter printer, Soroc IQ 120 terminal, and Micro Age Autoscribe word processor. Total price, $17,500.

for each character on the line. These printers are most often available for 150 lines per minute (lpm), 300 lpm, or 600 lpm. At 600 lpm the printer is usually waiting for the computer to catch up with it in micro and mini installations. A 600 lpm printer can range in cost from $5500 to more than $15,000 in today's market.

The reason why printer speed is so important, especially in the business environment, is that lengthy reports can tie up the computer for hours, in which case you are "printer-bound." It is important to consider what the printer will have to do, so that the installation can function efficiently. Unfortunately, it is common to discuss a problem with a business client, only to find out that he has added a program or two since designing his system. He now prints daily sales reports and inventory reports, and is startled to find that this can take several hours. Often, two printers, one for the high-volume internal distribution reports, and the other for the high-quality external reports, are required to overcome this problem.

For accounting, inventory, job-order control, or other complex jobs which require a medium to high volume of printed material or reports, a heavy-duty printer operating at 150 to 200 cps may be required. For a word-processing work station, a fully formed character printer (similar to a typewriter) is needed. The printer should run at 50 to 100 cps.

Finally, no machine provides instant answers to every question, no matter what the advertisements say. One of the biggest problems with the installation of any computer is that the expectations of the buyer usually far outreach the ability of the computer system (hardware, software, and operator) to meet those expectations. This problem reaches crisis proportions when the computer is sold or recommended by a nonprofessional. The well-informed small business user, who knows what he or she needs and what to expect, should do well in picking out a computer. And after you get your small business computer up and running, you'll wonder how you ever got along without it! ■

CENTRONICS COVERS THE COURT

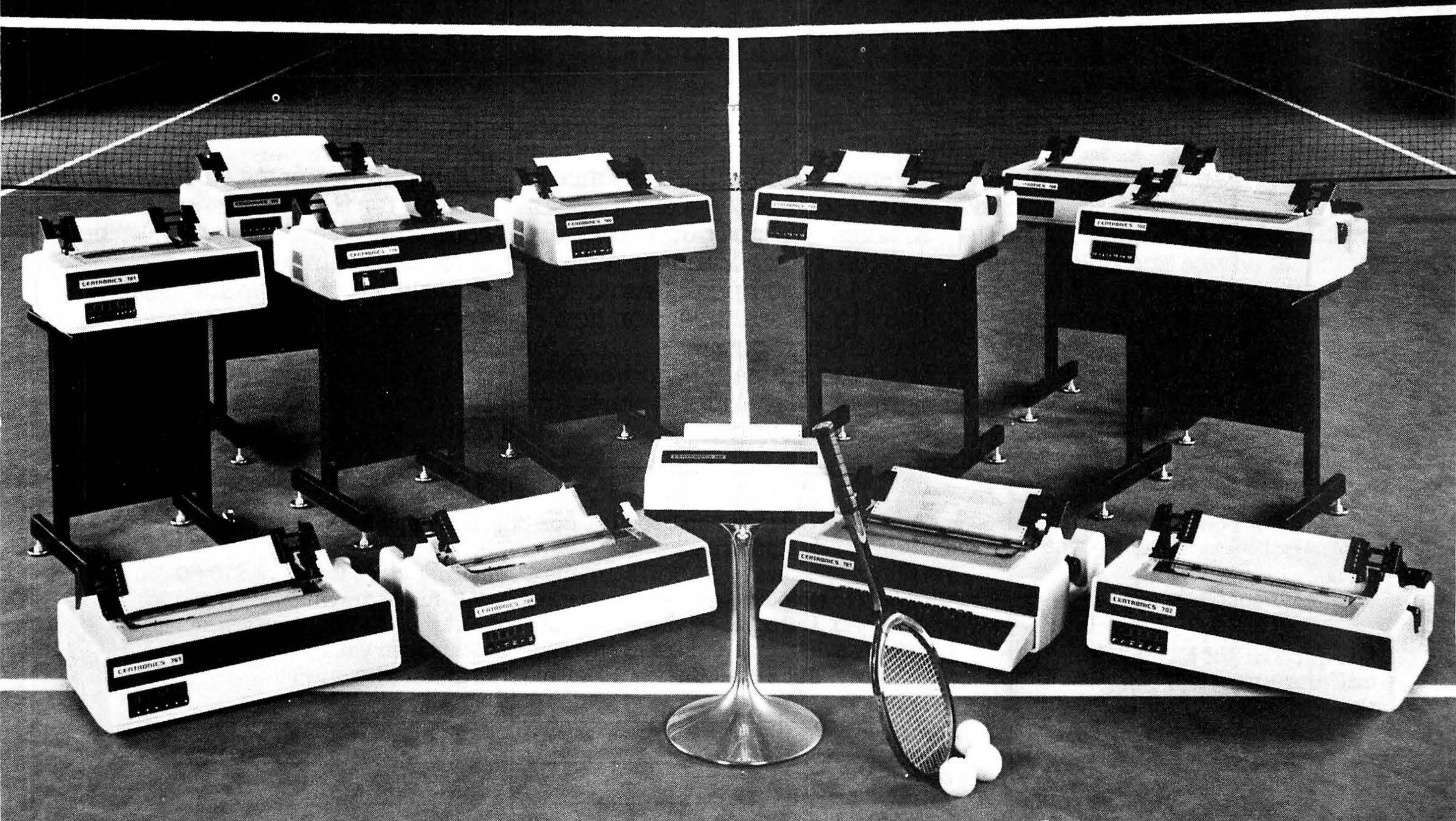
...with new, low-priced printers for small businesses

Now small businesses can have the advantage of Centronics performance. We have new models to meet the needs of small businesses— a selection that covers the court. And we've followed-through by pricing them **lower** than other printers that can't match Centronics' features and reliability.

TOP-RANKED TEAM We understand your small business needs—that's why Centronics has sold more printers to the small business market than anyone else. We have new, fully-featured models designed for small business applications. High throughput for inventory control. Full 132-column width for accounts receivable. Versatile forms handling capability for invoicing, payroll, and statements. Plus excellent print quality for labels and listings. The bottom line: with Centronics, small businesses can have mainframe performance at micro prices.

READY FOR ANY TOUR These printers are designed to deliver maximum in-service time, a key consideration for a small business. And we have the largest worldwide service organization of any independent printer company.

DON'T WRITE—phone Bob Cascarino today at (603) 883-0111, extension 4032, or contact any of our 15 U.S.A. or 9 international sales offices. Centronics Data Computer Corporation, Hudson, New Hampshire 03051.



CENTRONICS® PRINTERS
...the advantage

Circle 40 on inquiry card.

onComputing™
GUIDE TO PERSONAL COMPUTING

Name (Please print)

Street/Apartment Number

City | State/Province/Country | Postal Code

Please start my subscription with: ☐ Winter 1979 ☐ Spring 1980

☐ U.S. 1 yr. (4 issues) @$8.50 ☐ Canada & Mexico 1 yr. (4 issues) @$10.00

FOREIGN (to expedite service, please remit in U.S. funds drawn on a U.S. bank.)

☐ Europe (and all other countries, except above) 1 yr. @$12.00—surface delivery

☐ Bill VISA ☐ Bill Master Charge ☐ Bill me (North America only)

Card No. __________

Expiration __________

Four digits above name — Master Charge only __________

Signature __________ Date __________

Please allow eight weeks for processing. Thank you **4140**

onComputing™
GUIDE TO PERSONAL COMPUTING

Name (Please print)

Street/Apartment Number

City | State/Province/Country | Postal Code

Please start my subscription with: ☐ Winter 1979 ☐ Spring 1980

☐ U.S. 1 yr. (4 issues) @$8.50 ☐ Canada & Mexico 1 yr. (4 issues) @$10.00

FOREIGN (to expedite service, please remit in U.S. funds drawn on a U.S. bank.)

☐ Europe (and all other countries, except above) 1 yr. @$12.00—surface delivery

☐ Bill VISA ☐ Bill Master Charge ☐ Bill me (North America only)

Card No. __________

Expiration __________

Four digits above name — Master Charge only __________

Signature __________ Date __________

Please allow eight weeks for processing. Thank you **4140**

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS Permit Number 45 Martinsville NJ 08836

POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
POB 307
Martinsville NJ 08836
USA

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS Permit Number 45 Martinsville NJ 08836

POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
POB 307
Martinsville NJ 08836
USA

Some Advice From a Small Business Computer Expert

Photo by James Caccano

MicroAge's Jeff McKeever

MicroAge's Jeff McKeever of Phoenix, Arizona sells small business computers — a *lot* of them — and has been doing so for quite some time.

Jeff is chairman of The Phoenix Group, which operates under the name of MicroAge. The company is growing rapidly. It used to be primarily a supplier to the hobbyist and personal computer market, but Jeff's background in banking led him, along with president Alan Hald, to explore the small business market as well.

Today, a surprising number of business people are coming into MicroAge's stores (and other computer stores around the country) along with hobbyists, parents, students, and a host of other people. They are seriously evaluating the possibility of adding small computers to their businesses. Part of Jeff's philosophy is to offer the customer more than just a piece of hardware. The company offers a comprehensive guarantee with its hardware/software packages, since many small business customers do not have the time or inclination to maintain their systems when problems arise. Jeff feels that this is a vital key to success in the business market. In effect, he says, dealers have to try to offer the kind of service that IBM and other large companies have provided in the past. This is a tall order, but customers appear to be glad to pay the extra price to get the extra help.

In a recent interview with Jeff McKeever at his Tempe, Arizona headquarters, we asked him about his company and about small business computers in general.

onComputing: *When business people come to you with an interest in small business computers, but with no background in the subject, what do you tell them?*

Jeff McKeever: We strongly suggest that they get as much education as possible before they buy. One way to do this is by attending introductory classes, such as the ones we offer at MicroAge. Talking to colleagues who have installed computers and reading books are two other ways to get acquainted with the subject.

oC: *How much should a business spend for a small computer system?*

JM: Obviously every business is different from every other business in the amount of their gross margin, their need to automate, and so on. But as a general rule of thumb, a company should spend no more than two percent of its annual sales for hardware, and one percent for software. These guidelines should be applied judiciously to a given company's situation. A few years ago, before the advent of the microcomputer, the ratio of hardware to software was more like four to one for a minicomputer business system, but hardware costs have dropped significantly since then.

oC: *What about the hidden costs of computerizing a small business?*

JM: The real cost in adding a small business computer to your company is not in hardware or software — it's the cost of training your people to use the system. We see this in our own business, which is very automated. The real cost for us is in making sure that our people can understand and use the system.

oC: *Do you have any specific advice about getting your staff up to speed?*

JM: Yes. Do it slowly! Avoid the sense of future shock that can come from automating too rapidly. I've heard that some businesses actually go under because of a mad rush to automate. And don't throw out your existing manual systems too quickly. In our own business, a few months ago we were in the process of automating our accounts payable system. The system was performing so well that there was a temptation to discontinue the manual accounts payable process. I said no. We kept both the computerized and the manual systems going simultaneously for three months until I was absolutely sure that things were going smoothly. If you lose your accounting records during the transition and have no backup, you're dead. There's a tremendous overhead cost involved in taking this approach, but it's vital. A mistake the computer industry makes is in softselling the conversion process. In some cases, it could take a year or so to make the transition. ■

A SMALL BUSINESS COMPUTER DIRECTORY

Compiled by Charles Freiberg

Are you looking for a small business computer? Or perhaps some software? The following directory might be just what you are looking for. It lists all of the small business computer manufacturers known to us at this time, along with their hardware and software products.

The companies can be contacted directly, or you can visit your local computer store to see the equipment.

Every effort has been made to make this listing as complete as possible, but we realize that, inevitably, some errors might creep in. Blank categories in the directory indicate that information about a particular model was not available at press time.

We would appreciate hearing from readers about any errors or omissions to the listing. Corrections will be printed in the next issue of onComputing.

Business Hardware Directory

Key to Directory Categories:
Processor: The central "intelligence" of the computer, the place where data is processed, and where arithmetic and logical functions are performed. Many software packages for small business applications require a particular processor, so be sure to check this point with your computer dealer before you buy software *or* hardware.
Memory: The amount of storage capacity for programs and data in a computer. In this listing, "memory" refers to programmable memory, and is expressed in 8-bit bytes. Each byte of memory can hold one character (a letter, number, or symbol).
Available Software: Programs available from the manufacturer for use on a computer; *not* included in the indicated purchase price.
Price: Refers to suggested retail price. Prices may vary in different parts of the country, and many computer stores offer discounts on single units and package deals.
Standard Equipment: Peripherals and software included in the system package price.

Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
AlphaMicro Systems 17875 N Sky Park North Irvine CA 92714	Alpha Micro-systems	$2895	16-bit micro-programmed chip set	48 K		
Altos Computer Systems 2338-A Walsh Ave Santa Clara CA 95050	ACS-8000-7	$10,650	Z80	64 K	28 M byte Winchester drive with two platters, two double-density single-sided 8-inch floppy-disk drives (1 M bytes capacity). Six serial, two parallel I/O ports.	Software available: CP/M, CBASIC 2, MBASIC, Pascal, FORTRAN IV, COBOL, APL, Z80 macroassembler and C Compiler. Multiuser capability (up to four users) with AMEX multiuser operating system, up to 64 K for each user.
	ACS-8000-2	$4500	Z80	32 K	Two 8-inch single-sided single-density floppy-disk drives (½ M bytes capacity), two RS-232 I/O ports	
Apple Computer Inc 10260 Bandley Dr Cupertino CA 95014	Apple II Business System	under $5000	6502	48 K	9-inch video terminal, two 5-inch disks, Centronics 779 printer	Software: The Controller, General Ledger, Accounts Receivable and Payable, Account Aging, The Cashier, Apple Port, Dow Jones Series

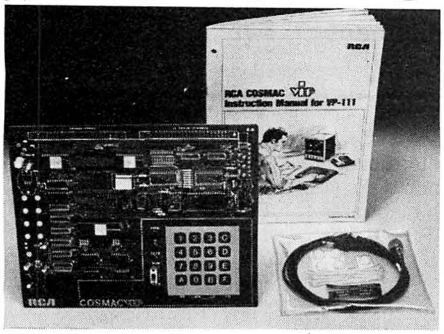
Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
Archives Inc 404 W 35th St Davenport IA 52806	Archives Business Computer	$6500	Z80	64 K	25 line by 80 character 12-inch video display terminal, serial and parallel I/O ports, dual 5-inch disks (372 K bytes)	Software: Word Processing, General Ledger, Accounts Receivable and Payable, Payroll, Inventory; Microsoft BASIC, FORTRAN, CBASIC 2
Associated Computer Industries 17952 Sky Park Cir Suite A Irvine CA 92714	ACI-90	$15,000	Microengine Pascal processor	64 K	video terminal, dual Shugart 8-inch disks, 16-bit Pascal Microengine, 2 M bytes storage, UCSD Pascal	Additional disks available for up to 4 M bytes of storage; General Ledger, Accounts Payable and Receivable, Payroll, Order Entry, Inventory
Basic Time 1215 E El Segundo Blvd El Segundo CA 90245	System B-100	$15,995	16-bit Data General Nova Instruction Set	64 K	24 line by 80 character video display terminal, one fixed and one removable hard disk (10 M bytes), Business BASIC; General Ledger, Accounts Payable and Receivable, Payroll, Inventory Control	Optional: printer with word-processing program
Cado System Corp 2730 Monterey St Torrance CA 90503	System 20/IV	$19,845	8085A	20 K expandable to 52 K	12-inch 24 line by 80 character video terminal, 100 cps bidirectional printer, 9 by 7 dot matrix, two dual-sided disks, four RS-232 I/O ports, furniture	Software: Billing, Accounts Receivable, Inventory, Payroll, Accounts Payable, General Ledger, Sales Accounting, Job Management, Purchase Order Writing, Optional hard disks and floppy disks
Carolina Business Computers Inc Oakwood Center 350 3rd Ave NW Hickory NC 28601	Miracle One	$15,950	8085	64 K	Soroc IQ 120 or Infoton VT-100 video display terminal, dual double-density 8-inch disks, TI-810 printer, S-100 mainframe, MBASIC, CP/M	Software: for vending-machine businesses
Columbia Data Products 9050 Red Branch Rd Columbia MD 21045	Commander 900 Commander 500	$4000	two Z80s	expandable to 80 K bytes	Video display, keyboard, optional built-in printer, CP/M, four RS-232 ports	Software available
Commodore Business Machines Inc 901 California Ave Palo Alto CA 94304	PET Business System (CBM)	$3585	6502	32 K	9-inch video terminal, dual disks (350 K bytes), 150 cps printer (tractor feed); file management in operating system	Software available: Word Processors, Real Estate Statistics, Banking and Finance, Mail List, Data Base Management, Payroll, General Ledger, Small Business Package, Inventory Control
Compal Computer Systems 12321 Ventura Blvd Studio City CA 91604	Compal 8100 Compal 8200	$8775 $10,995	Z80 Z80	56 K 56 K	12-inch video display, matrix printer plus software: BASIC and assembly, desk, manuals	General Ledger, Inventory, Accounts Receivable and Payable, Payroll, Order Entry, Sales Analysis, Data Management
COMPCO 8705 N Port Washington Rd Milwaukee WI 53217	ADI/OS	$9995	Z80	64 K	Microterm Mime-1 video terminal, Altos 8000-2 microcomputer, Houston Instruments HIPLOT, two double-density Shugart 5-inch disks, four RS-232 serial ports CP/M, FORTRAN, graphics software.	

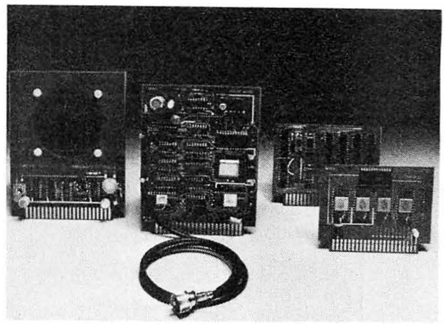
Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
Computer Data Systems Building 3 Drummond Pl Newark DE 19711	Versatile 4 Versatile 5	$5500 $4500	8085 8085	32/64 K 32/64 K	24 line by 80 character video display, quad- density dual 5-inch disks, keyboard	Software available: General Ledger, Accounts Receivable and Payable, Inventory Control, Order Entry, Payroll
Computhink Computer Corp 3260 Alpine Rd Menlo Park CA 94025	Minimax I	$4495	6502 (hybrid)	48 K	12-inch video terminal, 800 K bytes storage on two 8-inch disks, BASIC, serial RS-232 and parallel I/O ports	Optional: Printer, Business Software
Cromemco Inc 280 Bernardo Ave Mountain View CA 94040	CS-5	$5990	Z80A	32 K	S-100 bus, 21 board slots, two-disk and four-disk capacity, serial interface I/O card	80 character by 24 line video display terminal, line printer, letter printer, dual-disk drive, hard disk
Digital Equipment Corp Marlborough MA 01752	Datasystem 150	$12,050	PDP-11	32 K or 64 K	24 line by 80 character video terminal, dual 8-inch disks with 512 K bytes, operating system software, fully expandable and compatible with DEC COS-300 family	Software: in DIBOL business-oriented language
	WD-78 Business Accounting and Word Processing Computer	$12,495	LSI-8	32 K	LA 78P high-speed printer, dual 8-inch disk drives, minidesk, documentation. Computer has video display and keyboard.	Additional Software Available: Word Processor, General Ledger, Accounts Receivable, Inventory, Accounts Payable, Payroll ($400 each)
Digital Microsystems 4448 Piedmont Ave Oakland CA 94611	DSC-2	$4995	Z80	32 K standard 64 K optional	Double-density 5-inch disks direct memory access (DMA) disk controller, four RS-232 I/O ports, 16-bit parallel printer port	Optional: 8-inch disk (27.4 M bytes) Software available
Durango Systems Inc 3003 N First St San Jose CA 95134	F85	$13,385		65 K	9-inch video display, 16 by 64 or 24 by 80; 160 cps matrix printer, two 5-inch disks	Optional: two 5-inch disk drives (946 K) $3085; Fixed-disk drive (12 M bytes) $8990; Software available
Dynabyte Inc 115 Independence Dr Menlo Park CA 94025	DB 8-1	$2595	Z80	48 K 65 K optional	Two serial RS-232 or 20 mA ports, 12-slot backplane, eight levels of interrupt, jump start, ten interval timers.	
	DB 8-2	$4595	Z80	48 K 65 K optional	Same as above, plus: CP/M, double-density disk controller, single or double 5-inch disks (1.2 M bytes)	
Exidy Inc 390 Java Dr Sunnyvale CA 94086	Sorcerer	$1295	Z80	32 K (business system)	12-inch video display with graphics, serial and parallel interface, BASIC	Optional: printer, dual 5-inch disk cassette; software available, memory expansion capability
General Robotics Corp 57 N Main St Hartford WI 53027	Polaris	$12,000	LSI-11	64 K		
Heath Co Benton Harbor MI 49022	H-89 Kit	$1595	Z80 (2)	16 K expandable to 48 K	12-inch 25 line by 80 character video terminal, one 5-inch floppy disk, two serial ports, cassette interface, printer interface	Optional: two 5-inch disk drives, memory boards, printers, cassette players
Assembled version (Z89) marketed by Zenith Data Systems Inc	Z89 assembled	$2295				

Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
Hewlett-Packard 3400 E Harmony Rd Ft Collins CO 80525	HPZ50	$23,500	16-bit	160 K	24 line by 80 character video display terminal, double-density two-sided disks (1.2 M bytes), asynchronous serial interface, operating system, BASIC, 180 cps matrix printer	Software available Optional: 20 M byte fixed disk printer, remote consoles
Industrial Micro Systems 628 N Eckhoff St Orange CA 92668	Series 5000 Series 8000	$2800 $4300	Z80 Z80	32 K 32 K	500 K byte floppy-disk storage, two serial I/O ports, CP/M, one parallel I/O port	Two 5-inch double-density drives; two 8-inch double-density drives
INFO 2000 Corp 20620 S Leapwood Ave Carson CA 90746	INFO 2000	$15,800	Z80	40 K	video terminal, two-sided double-density dual 5-inch disks, word processing, line printer	Software: Data Base Manager, Word Processor, General Ledger, Accounts Receivable and Payable, Billing
Interactive Computer Systems 312 E 23rd St New York NY 10010	MCM System 908	$9500	AMD-2401	8 K expandable	12-inch video display, editing on-screen, terminal, APL, virtual memory (250 K bytes), RS-232 ports	Optional: dual 5-inch disks, hard disk system, printers Software available: accountants write-up, data-base management, word processor, order processing/inventory, business accounting system
IPEX International Inc 16140 Valerio St Van Nuys CA 91406	IPEX 8085	$3695	8085A	32 K	video terminal, dual 5-inch disks (600 K bytes), disk extended BASIC, dynamic debugging	Commercial software available: CP/M, CBASIC, Microsoft BASIC, FORTRAN 80, data-base management
Micro Application Systems Inc 5575 N County Rd Minneapolis MN 55442	MASEDT	$4799	6801 8086	48 K	5-inch video terminal, one or two 5-inch disks (80 thru 160 K bytes), 40- or 80-column dot-matrix printer, modem, BASIC compiler	
MicroDaSys POB 36051 Los Angeles CA 90036	System-Z	$2899	Z80	32 K	8-inch floppy disk, printer interface	Software: CP/M, Business Accounting, Inventory, Mailing List, Text Processor, BASIC compiler
Midwest Scientific 220 W Cedar Olathe KS 66061	MSI-6800 AW	$3895	6800B	32 K	two quad-density 5-inch disks, BASIC, serial interface, video terminal	Optional printers; Software available: Inventory Control, Accounts Payable and Receivable, General Ledger
North Star Computers 1440 4th St Berkeley CA 94710	Horizon	$2099	Z80 or 8080	32 K	one 5-inch disk drive	Options: single 5-inch disk ($600), two 5-inch disks ($1000), Pascal
Ohio Scientific 1333 S Chillicothe Rd Aurora OH 44202	C4P MF	$1695	6502	24 K	32 line by 64 character video display, one 5-inch disk	
Owl Electronic Labs Inc 233 Boston Post Rd Old Saybrook CT 06475	BMS-12 BMS-12FO BMA-E2M	$4695 $5790 $8445	Z80A Z80A Z80A	64 K 64 K 64 K	BMS-12: 12-inch video terminal, two 5-inch disks (640 K bytes) BMA-E2M: two 8-inch disks (2 M bytes)	Optional: Printers $450-$4950 DF2M (2 M byte disk) $3750

Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
Prodigy Systems Inc 497 Lincoln Hwy Iselin NJ 08830	Prodigy One	$12,500	Z80	32 K	24 by 80 video terminal, two 5-inch disks with 512 K bytes, 120-cps, 132-column printer, two business software packages, furniture	Software available: Accounts Receivable and Payable, Payroll, General Ledger, Inventory, Mailing List - each $500, Medical Billing - $2500, Order Entry - $2500
Psytek Inc 1900 Pickwick Ave Glenview IL 60025	Psytek 770	$9955	Z80A	48 K	30 line by 80 character video display, keyboard, 160-cps, 132-column matrix printer, two 8-inch disks	General Ledger, Accounts Receivable and Payable, Payroll, Inventory, CPA Management, Word Processing
Quay Corp POB 386 Freehold NJ 07728	Quay 900	$4000	Z80	48 K	two quad-density 8-inch disks (3.2 M bytes) CP/M, RS-232 or 20 mA serial port, Centronics-compatible printer interface	BASIC, FORTRAN, COBOL available
R2E of America 7 Bedford St Minneapolis MN 55414	Micral C	$8000	Z80	32 K	video terminal, two double-density 5-inch disks (320 K bytes), parallel printer interface	Software available Optional: General Ledger, Accounts Receivable and Payroll, Inventory
Radio Shack (Tandy) 1300 Tandy Center Fort Worth TX 76102	TRS-80 I	$849 (16 K byte version)	Z80	16 K expandable to 64 K	12-inch, 16 line by 64 character video display, cassette recorder	Optional hardware: printers, expansion interface, voice synthesizers, modems, 5-inch disk drives
	TRS-80 II	$3450	Z80A	32 K expandable to 64 K	12-inch, 24 by 80 character video display, detached keyboard, one 8-inch floppy-disk drive	Software available: Inventory Control System, Payroll, General Ledger, Accounts Receivable, Mailing List. Hardware available: disk expansion module, printers, furniture
Scientific Data Systems 12640 Beatrice St Los Angeles CA 90066	SOS-420	$7700	6502	32 K		
Smoke Signal Broadcasting 31336 Via Colinas Westlake Village CA 91361	Chieftain	$8500	6800 6809	40 K	24 line by 80 character video display, two 8-inch disks (500 K bytes), 80-column printer; Soroc terminal; Software included: Random-access DDS, Text Editor or processor, payroll processor	Software available: BASIC, COBOL, Invoice Entry, Inventory Control, Accounts Receivable, Order Entry, Payroll
Solid State Technology 17 Wheeling Ave Woburn MA 01801	Athena 8285	$3394	8085A	16 K expandable to 48 K	video terminal, RS-232 and 20 mA interface, AMOS, EDWIN (text editor)	Optional: built-in printers, two 5-inch disks, cassettes; Software available
Sord USA Inc International Trade Center 8300 NE Underground Dr Kansas City MO 64161	M200 Mark VI		Z80A	64 K	11.4 M byte Winchester hard-disk drive, video terminal	Software available: BASIC compiler, APL, FORTRAN IV, COBOL
Southwest Technical Products Corp 219 W Rhapsody San Antonio TX 78216	System B	$4995	6809	56 K	16 or 20 line by 80 character video terminal, dual 8-inch disks with dual-density controller (2.5 M bytes) (DMF-2)	Optional: capability to handle six terminals
	System D	$12,000	6809	128 K	three terminals, DMF-2 system, hard disk system (16 M bytes)	printers, 5-inch floppy-disk drives

The VIP hobby computer: Start programming for only $99.





New! VP 111 Microcomputer.... $99. Assembled* and tested.

Features:

- RCA 1802 Microprocessor.
- 1K Bytes static RAM.

Expandable on-board to 4K.
Expandable to 32K Bytes total.

- 512 Byte ROM operating system.
- CHIP-8 interpretive language or machine language programmable.
- Hexidecimal keypad.
- Audio tone generator.
- Single 5-volt operation.
- Video output to monitor or modulator.
- Cassette interface—100 Bytes/sec.
- Instruction Manual with 5 video game listings, schematics, CHIP-8, much more!

Ideal for low-cost control applications.

Expandable to full VIP capability with VP-114 Kit.

*User need only connect cables (included), a 5-volt power supply, and speaker.

New low price! The original VIP.. $199. Completely assembled and tested.

All the features of the VP-111 plus:

- A total of 2K Bytes static RAM.
- Power supply.
- 8 Bit input port.
- 8 Bit output port.
- I/O port connector.
- System expansion connector.
- Built-in speaker.
- Plastic cover.

Three comprehensive manuals:

- VIP Instruction Manual—20 video game listings, schematics, much more.
- VIP User's Guide—operating instructions and CHIP-8 for the beginner.
- RCA 1802 User's Manual (MPM-201B) - complete 1802 reference guide.

RCA

COSMAC VIP lets you add computer power a board at a time.

With easy-to-buy options, the versatile RCA COSMAC VIP means even more excitement. More challenges in graphics, games and control functions. For everyone, from youngster to serious hobbyist.

Built around an RCA COSMAC microprocessor, the VIP is easy to program and operate. Powerful CHIP-8 interpretive language gets you into programming the first evening. Complete documentation provided.

Send the coupon now...
Complete the coupon below and mail to: RCA VIP Customer Service, New Holland Avenue, Lancaster, PA 17604.

Or call toll free (800) 233-0094 to place your Master Charge or VISA credit card order. In Pennsylvania, call (717) 397-7661, extension 3179.

Please send me the RCA COSMAC VIP items indicated.

- ☐ **VP-111** New low cost Microcomputer (See description above) $99
- ☐ **VP-114** Expansion Kit for VP-111—Includes 3K RAM, I/O Port and connectors... $ 76
- ☐ **VP-711** VIP—The original VIP Microcomputer (See description above) $199
- ☐ **VP-44** RAM On-Board Expansion Kit—Four 2114 RAM IC's. Expands VP 711 memory to 4K bytes .... $ 36
- ☐ **VP-590** VIP Color Board—Converts VIP to color. Four background and eight foreground colors .... $ 69
- ☐ **VP-595** VIP Simple Sound Board—Provides 256 programmable frequencies. For simple music or sound effects. Includes speaker .... $ 30
- ☐ **VP-550** VIP Super Sound Board—Turns your VIP into a music synthesizer! Two independent sound channels. On-board tempo control. Outputs to audio system .... $ 49
- ☐ **VP-551** 4-Channel Super Sound—Includes VP-576 expander, demo cassette and manual. Requires VP-550 and 4K RAM .... $ 74
- ☐ **VP-570** VIP Memory Expansion Board—Plug-in 4K RAM memory .... $ 95
- ☐ **VP-580** VIP Auxiliary Keypad—Adds two-player interactive capability. 16-key keypad with cable. Connects to sockets on VP-590 or VP-585 .... $ 20
- ☐ **VP-585** VIP Keypad Interface Board—Interfaces two VP-580 Auxiliary Keypads to VIP .... $ 15
- ☐ **VP-560** VIP EPROM Board—Interfaces two 2716 EPROMs to VIP .... $ 34
- ☐ **VP-565** VIP EPROM Programmer Board—Programs 2716 EPROMs. With software .... $ 99
- ☐ **VP-575** VIP Expansion Board—Provides 4 buffered and one unbuffered expansion sockets .... $ 59
- ☐ **VP-576** VIP Two-Board Expander—Allows use of 2 Accessory Boards in either I/O or Expansion Socket .... $ 20
- ☐ **VP-601** ASCII Keyboard—128-character ASCII Encoded alphanumeric keyboard .... $ 65
- ☐ **VP-611** ASCII/Numeric Keyboard—Same as VP-601 plus 16 key numeric keypad .... $ 80
- ☐ **VP-620** Cable: Connects ASCII keyboards to VIP .... $ 20
- ☐ **VP-700** VIP Tiny BASIC ROM Board—BASIC code stored in 4K of ROM .... $ 39
- ☐ **VP-710** VIP Game Manual—Listing for 16 exciting games .... $ 10
- ☐ **VP-720** VIP Game Manual-II—More exciting games (Available 2nd qtr. '80) .... $ 10
- ☐ **MPM-201B** CDP1802 User Manual—(Included with VP-711) .... $ 5
- ☐ Please send more information .... —

Enclosed is $________ for items checked plus shipping & handling charge of $3.00.
Add your state and local taxes $________ Total enclosed $________
I enclose ☐ check or ☐ money order or, charge my ☐ VISA/Bank Americard ☐ Master Charge.

Credit card account No.________
Master Charge Interbank No. ________ Expiration date:________

Signature (required for credit card orders): ________
Name (please type or print):________
Street address: ________ City: ________
State & Zip: ________ Telephone: ( ) ________

Make checks payable to RCA Corp. Prices and specifications are subject to change without notice.

Circle 60 on inquiry card.

Manufacturer	Model	Price	Processor	Memory in Bytes	Standard Equipment	Additional Hardware and Software Available
Systems Engineering Enterprises 1749 Rockville Pike Rockville MD 20852	System 6684	$3995	6809	48 K	eight vectored interrupts, two double-sided 8-inch disks (1.2 M bytes), BASIC, macro-assembler, four RS-232 I/O ports	Optional: Printers, Pascal, FORTRAN, COBOL, BASIC; Software available: General Ledger, Accounts Receivable and Payable, Inventory Payroll; packages for physicians, publishers, etc
	System 6784	$4995	6809	48 K	24 line by 80 character video terminal	
Tano Corporation 4301 Poche Court West New Orleans LA 70129	Outpost II	$2595	6800	32 K	24 line by 80 character video terminal, 5-inch disk, BASIC, FLEX 2.0, TSC BASIC	Software available
Vector Graphic 31364 Colinas Westlake Village CA 91361	MZ	$4750	Z80A	48 K	Two 5-inch disks (630 K bytes) MDOS	
	Memorite	$8950	Z80	48 K	video terminal, two 5-inch disks, daisy-wheel printer	
Wordstream Corp 300 E 44 St New York NY 10017	WS1000	$17,990		64 K	video terminal, two 8-inch disks, printer, word processor	Software available: accounting packages
Zeda Computers International 1662 W 820 N Provo UT 84601	Zeda 580	$6837	Z80A	65 K	25 line by 80 character video display, two double-density 5-inch disks (400 K bytes), two RS-232 serial I/O ports, two parallel ports, parallel printer port, one hard-disk port, CP/M, ZEDOS, choice of one software package	Software available: Word Processor, Inventory, Accounting, General Ledger, Inventory

Business Software Directory

Key to Directory Categories:
Price: Refers to suggested retail price. Discounts may be available. Prices were correct at press time.
Medium: Refers to the physical form of the software as it is sold to the customer. Software in this listing is available on 5¼-inch floppy disks, 8-inch floppy disks, and/or audio-cassette tape.
Note: CP/M refers to a popular disk operating system for use with 8080 and Z80 computer systems. **CBASIC** is a form of BASIC for CP/M systems.

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	8-inch floppy disk	tape	PRICE	COMMENTS
Aaron Associates Inc POB 170A Garden Grove CA 92640	General Ledger, Inventory, Mail List, Payroll, Accounts Receivable, Accounts Payable, Cash Disbursements		X		$1595 complete package $320 each	
Advanced Information Design POB 2144 San Jose CA 95109	Accounts Receivable/Billing (balance forward, addresses, charges, credit limits, aged accounts, cash receipts, sales journal, open item data)		X			for the A.I.D. Business Computer System

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	8-inch floppy disk	tape	PRICE	COMMENTS
AJA Software POB 2528 Orange CA 92669	General Ledger, Accounts Receivable and Payable, Payroll, Inventory, Business Statistics, Letter Writer	X			$250	for North Star and TRS-80 computers
	Accounts Receivable	X			$50	for Apple II
American Business Systems Inc 9 Goldsmith St Littleton MA 01460	General Ledger, Accounts Payable, Accounts Receivable, Payroll, Order Entry, Inventory Control, Real Estate Automated Listing System		X		$1000 each	written in COBOL for Z80 based computers; requires printer and floppy-or hard-disk drive
Apple Computer Co 10260 Bandley Dr Cupertino CA 95014	The Controller General Ledger Accounts Receivable and Payable Account Aging The Cashier Dow Jones Series	X X X X X X				for use with Apple II computer
Arkansas Systems Inc 8901 Kanis Rd Suite 206 Little Rock AR 72205	General Ledger, Accounts Receivable, Accounts Payable, Order Entry, Inventory, Payroll	X			$795 GL $495 AR $495 AP $795 P	written in FORTRAN for CP/M systems
A-T Enterprises 221 N Lois St La Habra CA 90631	Income Property Management (income and expenses, accounting control, budgeting, list of tenants, charges, property analysis, etc)		X		$750	written in CBASIC for 8080 or Z80 based computers with 48 K bytes of memory, two 8-inch drives, and CP/M; handles seventy-five properties and 1500 units
Better Programming Systems 275 Fort Washington Ave New York NY 10032	Accounting, Payroll, Check Printing, Invoicing, Inventory, Time Analysis		X		$500	for 48 K Ohio Scientific Challenger III with 1 M bytes of memory and printer
The Bottom Shelf Inc POB 49104 Atlanta GA 30359	The Business Mailing System (holds 150,000 names in ZIP code order and alphabetical order)	X			$125	for TRS-80 computer with 32 K bytes of memory, printer, and two 5-inch disk drives; each disk holds 500 names, uses one thru four labels at a time
Business Application Software 16755 Littlefield Ln Los Gatos CA 95030	General Ledger Accounts Receivable Accounts Payable		X X X		$250 $90 $90	requires CBASIC and CP/M
California Business Computers Corp 825 W Hamilton Ave Campbell CA 95008	Earned Income Payroll with earned income credit provisions	X	X		$595	requires CP/M system CBASIC or CBASIC 2
Commercial Computer Inc 9742 Humboldt Ave S Minneapolis MN 55431	General Ledger, Accounts Receivable, Accounts Payable, Inventory, Mail List, Payroll	X			$100 each	
Computer Custom Systems Inc 603 W 13th St Austin TX 78701	General Ledger, Accounts Receivable and Payable, Payroll, Inventory, Job Cost Analysis	X			$200 GL $200 AR $200 AP $175 P $175 I $175 JCA	for TRS-80 with minimum of two 5-inch disks

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	8-inch floppy disk	tape	PRICE	COMMENTS
Contex Systems 1340 Galindo St Concord CA 94520	PM (preventive maintenance - automatic equipment maintenance scheduling)	X			$1000	written in CBASIC for 8080 or Z80 based computer with 48 K bytes of memory, dual disk CP/M
Creative Discount Software 256 S Robertson Suite 2156 Beverly Hills CA 90211	New Stat Pac Master Text Project Manager System	X X X			$70 $120 $45	Apple II and TRS-80
Cybernetics Inc 8041 Newman Ave Suite 208 Huntington Beach CA 92647	General Ledger Accounts Receivable and Payable Payroll with cost accounting		X X X		$250 $250 $250	for TRS-80 computer with CP/M, CBASIC 2
Dr Daley 425 Grove Ave Berrien Springs MI 49103	Inventory, Estimate, and Mail List	X			$99.95 each	for PET computer
Data Access Corp 11205 S Dixie Hwy Miami FL 33156	Inventory Accounts Receivable General Ledger Payroll Mailing List		X X X X X		$600 $500 $500 $500 $200	written in BASIC
Delta Data Systems 361 Parkway Dr Pocatello ID 83201	Deltax Income Tax Preparation System (form 1040, 2210, 2440, 2441, 3903, 4625, 4726, schedules A,B,C,D, E,F,G,R,RP,SE,TC) Tax Audit	X X			$995 license fee $55	written in North Star BASIC
Digital Data Systems Inc 1396 NW 65th Ter Plantation FL 33313	General Ledger Accounts Receivable		X X		$995 $895	written for Cromemco System 3 computer
Ellis Computing 1480 17th Ave San Francisco CA 94122	Budget Planning Personal Financial Reporting Labels	X X X			$25 $25 $25	written in Nevada COBOL; runs with any COBOL compiler
G W Computers Ltd 89 Bedford Court Mansions Bedford Ave London W1 ENGLAND	CP/M Business Package (General Ledger, Accounts Payable and Receivable, Inventory, Order Entry, etc)	X			$750	for PET 32 K disk system SwTPC 6800, Apple II, and TRS-80 Level II computers; distributed in US by John D Owens Associates, 12 Schubert St, Staten Is, New York NY 10305
H Geller Computer Systems POB 350 New York NY 10040	Tape Data Query (Inventory, Accounts Receivable and Payable, Order Entry, General Ledger)			X	$100	for PET 8 K and TRS-80 Level II computers; requires two cassette recorders
Global Parameters 1505 Ocean Ave Brooklyn NY 11230	Database Management System		X		$295	runs under CP/M and CBASIC with minimum of 40 K bytes of memory
Graham-Dorian Software Systems 211 N Broadway Wichita KS 67202	Payroll System Apartment Management System Inventory System Cash Register System		X X X X		$590 $590 $590 $590	requires CBASIC Distributor: Lifeboat Associates 2248 Broadway New York NY 10024

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	 8-inch floppy disk	 tape	PRICE	COMMENTS
Information Access Systems Inc 1129 Bloomfield Ave W Caldwell NJ 07006	General Ledger, Accounts Receivable, Accounts Payable, Inventory, Payroll, Order Processing		X			for Heath WH11A computer
International Micro Systems Inc 3077 Merriam Ln Kansas City KS 66106	General Ledger		X		$750	other products available
	Accounts Receivable		X		$750	
	Inventory		X		$350-$950	
	Payroll		X		$750	
	Mail List		X		$100	
	Cash Receipts		X		$150	
Johnson Associates POB 1402 Redding CA 96001	Index Sequential Access Method (controls business application files on disk)	X			$50	for TRS-80 computer
	Transaction Data Entry (handles business application keyboard data entry)	X			$20	
Kenmor Co 675 McLean Ave Yonkers NY 10704	Accu-Tax (income tax program for all states)		X		$995	for Z80, 8080 or any CP/M based system with 32 K bytes of memory and 132-column printer; written in CBASIC
Mad Hatter Software 900b Salem Rd Dracut MA 01826	Client Billing, Accounts Receivable, Accounts Payable, General Ledger, Inventory, Payroll and Mailing List, File Handling (for TRS-80)	X			$795 Apple up to $995 TRS-80	
Charles Mann and Associates 7594 San Remo Trl Yucca Valley CA 92284	Project Management System (budgeting, cost recording, variance reporting, completion cost estimating)	X			$69.95	designed for TRS-80 computer with a 5-inch floppy disk
MicroAge 1425 W 12th Pl Tempe AZ 85281	Autoscribe (Word Processor)		X		$395	Unless otherwise indicated, all software is for use on North Star computer systems. Autoscribe also available for Heath/Zenith WH89 computer. Autoscribe and Moneybelt also available on CP/M
	Timekeeper (Time and Accounting Billing System)		X		$595	
	Moneybelt (Accounts Receivable and Payable, and General Ledger)		X		$495	
	Ledger Plus (Apple)	X			$295	
	Ledger Plus (Vector Graphic)		X		$495	
	CPA Client Writeup		X		$995	
Micro-Ap 9807 Davona Dr San Ramon CA 94583	Selector III Information Management System		X		$345	requires 52 K bytes of memory CP/M, CBASIC 2 and Micro-Ap Selector III-C2 system
Micro Architect 96 Dothan St Arlington MA 02174	ACCT-III (Accounts Receivable, Initialization, Account Manager, Report Generator)		X		$69	written in BASIC for TRS-80; requires two 5-inch floppy disks, 32 K bytes of memory, and an 80-column printer
	Inventory System		X		$149	
	Mailing List		X		$99	
Micro Data Base Systems Inc POB 248 Lafayette IN 47902	Data base management system (DBMS)	X	X		$750	contains data definition language analyzer/editor, data management routines, sample applications programs, and user's manual

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	 8-inch floppy disk	 tape	PRICE	COMMENTS
Micro Mike's Inc 905 S Buchanan Amarillo TX 79101	General Ledger Accounts Receivable Accounts Payable Payroll Inventory	X X X X X			$49.95 each	written in BASIC with CSUB for North Star disk system
Microcomputer Applications Inc 4614 Trail Crest Cir Austin TX 78735	Business and Financial Analysis Program (portfolio, home ownership, economic order quantity, etc)	X		X	$36	for TRS-80 Level II and CP/M, also for Kansas City; requires 10 K bytes of storage
Microcomputer Consultants POB 255625 Sacramento CA 95825	Inventory Control for Manufacturing (inventory listing by category and with valuation, assembly and job issue listing, job status report, costed bill of materials)		X		$1250	written in CBASIC; runs on dual-drive CP/M system with 48 K bytes of programmable memory
Micropro International 5810 Commerce Blvd Rohnert Park CA 94928	Mail List		X		$95	for CP/M systems
Midwest Computer Peripherals 1467 S Michigan Ave Chicago IL 60605	General Ledger Accounts Receivable Accounts Payable Payroll Inventory II	X X X X X			$95 $95 $95 $95 $99	for TRS-80
National Software Exchange Inc 1000 Lake Saint Louis Blvd Suite 17 Lake Saint Louis MO 63367	Accounts Receivable, Accounts Payable, General Ledger, Inventory, Payroll		X		$850 for the package	for CP/M systems
National Software Marketing Inc 4701 McKinley St Hollywood FL 33021	Project Management System Time Accounting System Inventory Control System	X X X			$116 $66 $89.95	designed for TRS-80 computer with 32 K bytes of memory and 5-inch floppy disks
Osborne/McGraw-Hill Inc 630 Bancroft Way Dept 126 Berkeley CA 94710	Payroll, Accounts Payable and Receivable, General Ledger	X			$15 each	written in CBASIC
Peachtree Software Retail Sciences Inc Suite 419 3384 Peachtree Rd Atlanta GA 30326	Integrated Accounting Software System (General Ledger, Accounts Receivable, Accounts Payable, Payroll)		X		$1000 each	written in Microsoft BASIC for CP/M, using 8080-compatible system with 48 K bytes of memory and printer
Percom Data Co Inc 211 N Kirby Garland TX 75042	General Ledger System Finder (Data Base Manager) Mailing List Processor	X X X			$199.95 $99.95 $99.95	for the Percom LFD-400 disk-drive system
Personal Software Inc 592 Weddell Dr Sunnyvale CA 94086	VisiCalc visual display system for business problem solving	X			$150	for Apple II; system allows user to manipulate and change data on the screen; has calculator like features.
	Data Management System	X				for TRS-80
PS Inc 619 N P Ave Fargo ND 58107	P.S. Accounting Package (General Ledger, Accounts Payable, Accounts Receivable, Order Entry, Inventory Control)	X			$65	runs on PDP-11, AM-100, Altos computers and Pascal Microengine

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	8-inch floppy disk	tape	PRICE	COMMENTS
Racet Computes 702 Palmdale Orange CA 92665	Infinite BASIC with Business Package add-on	X			$79.90	enhanced BASIC with advanced matrix manipulation, multiple precision arithmetic, etc
Realty Software Co 2045 Manhattan Ave Hermosa Beach CA 90254	Tax-Deferred Property Exchange Model	X		X	$25 disk $20 cassette	written for Apple II and TRS-80 Level II
Rothenberg Information Systems Inc 260 Sheridan Ave Palo Alto CA 94036	General Ledger, Accounts Receivable, Accounts Payable, Payroll, Inventory		X		$500 each	requires CBASIC under CP/M
Scot-Ware POB 430734 7000 SW 62nd Ave Miami FL 33143	F3 Calculator Program (calculates life cycle savings and payout time on investment; figures inflation rates on energy, property taxes, and operating costs)				$95	magnetic card program for TI-59 calculators
		X			$120	Level II Microsoft BASIC for TRS-80
Small Business Computer Services 813 MacArthur Dr Urbana IL 61801	General Ledger, Accounts Receivable, Accounts Payable	X			$250	written for systems with dual 5-inch floppy disk Micropolis Mod II with 48 K bytes of memory using 8080 and Z80 and a printer
Smoke Signal Broadcasting 31336 Via Colinas Westlake Village CA 91361	Single-User COBOL					for use with Smoke Signal Chieftain computer with 48 K bytes of memory and 500 K bytes of disk storage
	Development system		X		$3960	
	Runtime Package		X		$595	
	Multi-User COBOL					
	Development system		X		$4950	
	Runtime Package		X		$745	
	Auto Clerk		X		$1320	
	Auto Index		X		$1320	
	Accounts Receivable		X		Prices	
	General Ledger		X		determined	
	Payroll		X		by dealers	
	Inventory		X			
Software Consultation Design and Production (SCDP) 6542 Greeley St Tujunga CA 91042	Vulcan data base management system		X		$490	for CP/M or Processor Technology PTDOS systems
Software Exchange 17 Briar Cliff Dr Milford NH 03055	Inventory S			X	$25	for TRS-80 with 32 K bytes of storage
	Inventory II.2	X			$59.95	
	Inventory System 2.3	X			$99.95	
	Accounts Receivable	X			$79.75	
	General Ledger	X			$79.95	
	Small Business Bookkeeping	X			$31.95	
	Payroll	X			$59.95	
Software Industries 902 Pinecrest Richardson TX 75080	Mailing List (add, delete, search, sorted list, modify address, sequential printout)	X			$39.95	designed for TRS-80 computer with one or two 5-inch floppy disks
Strictly Software 16720 Hawthorne Blvd Lawndale CA 90260	General Ledger, Accounts Receivable, Accounts Payable, Inventory, Payroll, Mail List, Sales Journal, Cash Disbursements, etc	X			$1750 for the complete package	for CP/M systems

MANUFACTURER	PROGRAM	SOFTWARE MEDIUM 5-inch floppy disk	8-inch floppy disk	tape	PRICE	COMMENTS
Structured Systems Group 5204 Claremont Oakland CA 94618	General Ledger Accounts Receivable Accounts Payable		X X X		$899 $699 $699	requires CBASIC; Distributor: Lifeboat Associates 2248 Broadway New York NY 10024
Supersoft POB 1628 Champaign IL 61820	Client Record System		X		$250	written for North Star systems; requires one disk
Technical Software Inc POB 73043 Metairie LA 70033	Investment and Mortgage Management	X			$17.50	for Apple II computer; minimum of 8 K bytes of memory needed
Thorstensen Labs 66 Littleton Rd Westford MA 01886	Mailing List Payroll Inventory	X X X	X X X		$75 $150 $150 $250 $150 $250	designed for Tano Corp Outpost 11 computer with 48 K bytes of memory and two 5-inch floppy disks
Timberline Systems Inc 10550 SW Allen Blvd Suite 114 Beaverton OR 97005	Property Management System		X		$3200	recommended for systems having a hard-disk drive
United Software of America 750 3rd Ave New York NY 10017	Accounts Receivable/Payable Cash Receipts and Disbursements	X X			$175 $99.95	for PET computer with 32 K bytes of memory, dual 5-inch disk drive, and printer

FOR MORE INFORMATION

on the Companies listed in our Small Business Computers Directory, please refer to the list below and circle the appropriate Reader Service number on the inquiry card at the back of this magazine.

Hardware

R.S.#	Advertiser
1	AlphaMicro Systems
2	Altos Computer Systems
3	Apple Computer Inc.
4	Archives Inc.
5	Associated Computer Industries
6	Basic Time
7	Cado System Corp.
8	Carolina Business Computers Inc.
9	Columbia Data Products
11	Commodore Business Machines Inc.
12	Compal Computer Systems
13	COMPCO
14	Computer Data Systems
15	Computhink Computer Corp.
16	Cromemco Inc.
17	Digital Equipment Corp.
18	Digital Microsystems
19	Durango Systems Inc.
21	Dynabyte Inc.
22	Exidy Inc.
23	General Robotics Corp.
24	Heath Co.
25	Hewlett-Packard
26	Industrial Micro Systems
27	INFO 2000 Corp.
28	Interactive Computer
29	IPEX International Inc.
31	Micro Application Systems Inc.
32	MicroDaSys
33	Midwest Scientific
34	North Star Computers
35	Ohio Scientific
36	Owl Electronic Labs Inc.
37	Prodigy Systems Inc.
38	Psytek Inc.
39	Quay Corp.
41	R2E of America
42	Radio Shack (Tandy)
43	Scientific Data Systems
44	Smoke Signal Broadcasting
45	Solid State Technology
46	Sord USA Inc.
47	Southwest Technical Products Corp.

Software

R.S.#	Advertiser
51	AJA Software
52	American Business Systems Inc.
53	Apple Computer Co.
54	Arkansas Systems Inc.
55	A-T Enterprises
56	Better Programming Systems
57	The Bottom Shelf Inc.
58	Business Application Sftwre.
59	California Business Computers Corp.
61	Commercial Computer Inc.
62	Computer Custom Systems Inc.
63	Contex Systems
64	Creative Discount
65	Cybernetics Inc.
66	Dr. Daley
67	Data Access Corp.
68	Delta Data Systems
69	Digital Data Systems Inc.
71	Ellis Computing
72	G.W. Computers Ltd.
73	H. Geller Computer Systems
74	Global Parameters
75	Graham-Dorian Software Systems
76	Information Access Systems Inc.
77	International Micro Systems Inc.
78	Johnson Associates
79	Kenmor Company
81	Mad Hatter Software
82	Charles Mann and Associates
83	MicroAge
84	Micro-Ap
86	Micro Architect
87	Micro Data Base Systems Inc.
88	Micro Mike's Inc.
89	Microcomputer Applications Inc.
91	Microcomputer Consultants
92	Micropro International
93	Midwest Computer Peripherals
94	National Software Exchange Inc.
95	National Software Marketing Inc.
96	Osborne/McGraw-Hill Inc.
97	Peachtree Software Retail Sciences Inc.
98	Percom Data Co. Inc.
99	Personal Software Inc.
101	PS Inc.
102	Racet Computes
103	Realty Software Co.
104	Rothenberg Information Systems Inc.
105	Scot-Ware
106	Small Business Computer Services
107	Smoke Signal Broadcasting
108	Software Consultation Design and Production
109	Software Exchange
111	Software Industries
112	Strictly Software
113	Structured Systems Group
114	Supersoft
115	Technical Software Inc.
116	Thorstensen Labs
117	Timberline Systems Inc.
118	United Software of America

You Just Bought a Personal What?

by Thomas Dwyer and Margot Critchfield

Whether you are a novice programmer or an experienced computer user, this book is filled with practical ideas for using a personal computer at home or work. It will take you through the steps necessary to write your own computer programs, and then show you how to use structured design techniques to tackle a variety of larger projects. The book contains over 60 ready-to-use programs written in Radio Shack TRS-80 Level II BASIC in the areas of educational games, financial record keeping, business transactions, disk-based data file and word processing. $11.95 ISBN 0-07-018492-5

Please send ☐ ____________ copies of **You Just Bought a Personal What?**

Name ____ Title ____ Company ____

Street ____ City ____ State/Province ____ Code ____

☐ Check enclosed in the amount of $____________

☐ Bill Visa ☐ Bill Master Charge

Card No. ____________ Exp. Date ____________

Add 60¢ per book to cover postage and handling.

BYTE BOOKS 70 Main Street
Peterborough, New Hampshire 03458

Error Messages

Dear Editor:

I read with interest your article on the Heathkit H8. I was particularly impressed by the time to complete the kit as described on page 66 (Winter 1979).

"... in less than one hundred working hours spread over five or six days." Now let's see, 100 hours divided by 5 or 6 equals ....

Michael L Steiner
Rocky Mount NC

[Oops! Our computer dropped some bits on that one....ed]

Calculator Bug

Dear Editor:

I have recently read the article "Use Your Pocket Calculator to Save Money."

The program listed for use with a TI-59 programmable calculator seems to have some errors and omissions. There is an omission with respect to location 042. The program jumps from 041 to 043. The balance of the program does not "RUN" as per the narrative and table in figure 4.

The program has some nice objectives, and could be a very useful utility package.

Harold N Saphin
Roslyn NY

Unfortunately, four typographical errors—two of omission, two of commission—appeared in an article in the Winter 1979 onComputing, "Use Your Pocket Calculator to Save Money," by Louis Hohenstein, pages 12 thru 14 and 74 thru 76. The errors were not *the fault of the author, but rather resulted from an editorial gaffe.*

The first error of omission was in figure 6 on page 76, a listing of a program for the Texas Instruments TI-59 calculator. Line 042 of the program was left out. It should have appeared in the following sequence:

```
041  43  RCL
042  03  03
043  69  OP
```

The first error of commission also was in figure 6. At line 118, the instruction mnemonic should have been SBR, and not SUM as was shown.

The second error of omission concerned the procedure for repartitioning the memory space of the calculator prior to running the program. After you load the program, you must execute the following sequence of keystrokes to perform the partitioning operation. Press $\boxed{1}$, $\boxed{0}$, $\boxed{2nd}$, $\boxed{OP}$, $\boxed{1}$, *and* $\boxed{7}$ *in that order.*

Observe that registers used for storing amounts are the eighty-nine registers numbered 11 thru 99. Registers 00 thru 10 are used in the program for storing text and working totals.

The second error of commission was on page 14 in the first column. The display should have read:

```
1234.     CKNO
```

and not:

```
1234.     CKND
```

as was printed.

We apologize to readers who had difficulty in running the program....ed


POYNETTE
FAMILY PRACTICE CENTER

Profile:

The Small Town Doctor and the Personal Computer

Photos and Text by Christopher Morgan

A personal computer in a small town clinic is creating a quiet revolution in medical care.

Poynette, Wisconsin (population 2000) is a good place to get sick.

For one thing, the Poynette Clinic is run by a staff of dedicated, competent people. For another, their major assistant is a friendly, patient, and versatile personal computer.

The clinic's director, Dr Douglas McNeill, MD, is hooked on computers — so much so, in fact, that he missed his Phi Beta Kappa induction at Columbia University because he was playing tic-tac-toe with a computer! ("It's absolutely true," he said. "They called me up when it was over to find out what had happened to me. I had completely forgotten about it. It was one of the most embarrassing moments of my life!")

His dedication to computers began during his student days at the Columbia University College of Physicians and Surgeons. One day while he was hitchhiking to the Laundromat, Dr McNeill was picked up by a computer science professor who said he was looking for someone to run an IBM 1620 computer for a numerical analysis course he was giving. Dr McNeill expressed an interest in the job and was hired. This led him to further involvements. Classes in FORTRAN and BASIC fueled his burning desire to apply computer technology to his life's work. After graduation from medical school, he set up practice in a small town, away from the sometimes rigid framework of the urban or suburban health center.

Photo 3: The doctor's computer setup, including Heath LSI-11 computer and WH-27 dual floppy-disk drives. Not shown is the DECwriter used to print patient's bills and other information.

I visited the Poynette Clinic recently. Poynette is located in the rolling farm country of southern Wisconsin, thirty miles north of Madison. Once inside the building, I knew it was no ordinary clinic. In the corner sat a Heath LSI-11 computer with two Heath WH-27 floppy-disk drives, a DECwriter, and a Heath H9 video terminal. Someone at the terminal was keying in patient data.

The entire operation is computer-oriented. When a patient comes to the clinic, he or she is given a plastic identification card. The card is carried by the patient from station to station. All paperwork, charts, forms, prescriptions, and so on are stamped with the patient's data code number. These transactions are eventually entered into the patient's computer records.

The patient's medical history is also entered into the computer. The usual questions are asked, plus some unusual ones: Do you wear seat belts? Can you swim? If you are a woman, do you perform breast self-examination? Have you ever been exposed to asbestos? Have you ever had head or neck X-ray therapy?

Dr McNeill believes that such information is just as important as the patient's history of diseases, because it helps him to practice true preventive medicine. In some cases

the information is only of use *after* a patient's illness or accident, but it furnishes vital clues to the causes and effects in a chain of events, and ultimately makes the doctor wiser. As Dr McNeill frankly puts it, "I want to know what makes people die."

Today's doctor needs to know much more about the patient's environment, habits, and feelings than was the case in the past, in order to offer more effective health care. The personal computer can help do this because it never forgets. It is also affordable. But most important of all, it is adaptable to the will of the individual user — much more so than a large computer could ever be.

After data gathering comes data processing. One program (written by Dr McNeill, as were all of the programs he uses) tabulates the data about problems or ailments of a large group of patients and displays it in both percent and graph form (see figure 1). Health trends become more obvious by using this method. For instance, the chart reveals a much higher than normal incidence of asthma in the local population. In fact, the number of people who have asthma in Poynette is greater than the number of people who have high blood pressure (hypertension) — not usually the case in the general population. Some local factor is probably at work causing the respiratory condition. Although Dr McNeill has not yet isolated the factor, he at least knows about it thanks to the computer.

"I want to put everything I know about the patient on the computer," Dr McNeill said. "I pledge to keep all the information about the patient so I can grow wiser. If I make a mistake, I'll know why. And if I'm successful, I'll know why, too. Many of my questions are designed to find *potential* problems that might worsen if not caught in time."

Another of the doctor's programs can supply the names of patients who have various *combinations* of ailments. For example, a list of people who smoke and who also have high blood pressure can be provided. The doctor can then send a note to each patient warning of the possible hazards of lung cancer. In fact, the computer can automatically add such a note to the bottom of a patient's bill. (Billing, incidentally, is another function carried out by the computer.)

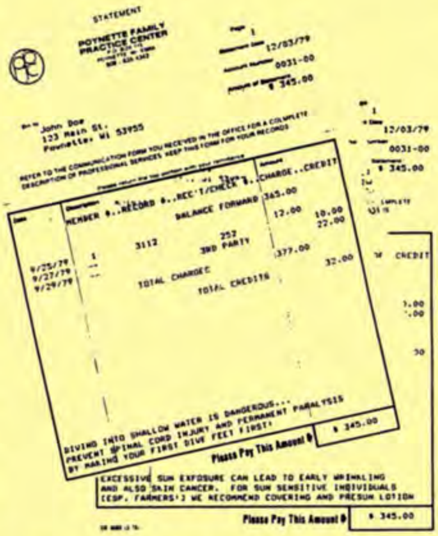

STATEMENT
POYNETTE FAMILY PRACTICE CENTER
12/03/79
0031-00
$ 345.00
John Doe
123 Main St.
Poynette, WI 53955
12/03/79
0031-00
$ 345.00
BALANCE FORWARD 345.00
3112
252 3RD PARTY
9/25/79
9/27/79
9/29/79
12.00
10.00
22.00
377.00
32.00
TOTAL CHARGES
TOTAL CREDITS
DIVING INTO SHALLOW WATER IS DANGEROUS... PREVENT SPINAL CORD INJURY AND PERMANENT PARALYSIS BY MAKING YOUR FIRST DIVE FEET FIRST!
Please Pay This Amount $ 345.00
EXCESSIVE SUN EXPOSURE CAN LEAD TO EARLY WRINKLING AND ALSO SKIN CANCER. FOR SUN SENSITIVE INDIVIDUALS (ESP. FARMERS!) WE RECOMMEND COVERING AND PRESUN LOTION
Please Pay This Amount $ 345.00

Photo 4: A typical computer-generated patient's bill. The doctor's computerized "bedside manner" is revealed in the personal safety message printed at the bottom of the bill.

Dr McNeill is young both in spirit and in age (he is 33 years old), and his conversation is a delightful mixture of candor and wit. His computerized bills are apt to contain informal messages to his patients, such as:

> SAFETY TIP — WE LOVE YOUR FINGERS AND TOES AS MUCH AS YOU DO. PLEASE USE YOUR ROTARY MOWER ONLY IN DRY GRASS WHILE WEARING SHOES, AND STOP IT TO FUEL UP OR CLEAN OUT CLOGS.

Behind the doctor's lightheartedness, however, is a fierce sense of responsibility toward his profession and toward the computer's role in that profession. Dr McNeill has recorded some of his beliefs in the Poynette Clinic's "Principles of Practice," which reads (in part) as follows:

"We believe that the medical record is held in trust for the patient by the provider. It is, therefore, our responsibility to maintain records scientifically and with a clarity that permits and encourages patient understanding of these documents. . .

"Patients shall be encouraged, but not required, to have a comprehensive defined data base within two years of entry into the practice. . .

"We acknowledge that patients are first of all people, and, as people, have the right and need to participate in decisions that affect them, especially those that affect them in as intimate a manner as health care. . ."

Some of Dr McNeill's ideas were inspired by the pioneering work of Dr Larry Weed, MD, at the PROMIS Laboratory of the University of Vermont. Like Dr McNeill, Larry Weed is convinced that the computer is the real key to modernizing medicine. Writing in *Modern Hospital* magazine, Dr Weed states:

"Up until now the human, operating at the patient's bedside and crudely replenished by random contacts with literature, has been the course of and has set in motion all the information energy that was directed to the patient's needs. The average physician's brain fell behind in every way and, worst of all, there was no central superbrain, complete and up-to-date, acting as a standard to which all could turn. . . The computer allows a speed and a multiplicity of correlations of action and knowledge never before attainable."

Much of Dr Weed's thought on the subject has been distilled in his book, *YOUR Health Care and How To Manage It,* a plea for the intelligent management of health care

both by the doctor and the patient. (Former *Saturday Review* editor Norman Cousins has also echoed the patient's vital role in the healing process in his recent book, *Anatomy of an Illness.*)

* * *

The modern idea of holistic medicine is an attempt to treat the entire person, both psychologically and physically. To do this, the physician requires data and more data.

Hundreds of physicians across the country and around the world have begun using personal computers in their practices. The idea of a central data base of medical information available to all physicians is a real possibility. Dr McNeill's voice is beginning to be heard, most recently in an article by him in the *Physician's Microcomputer Report.*

It takes the good humor and dedication of a Doug McNeill to spread the word about computers and medicine.

And who knows? Maybe someday you'll be cured by a physician wielding both a scalpel and a well-written subroutine.■

Identification Code	Medical Problem	Number of Cases	Histogram
278.0	Obesity	123	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX* 19.1%
305.1	Smoking	111	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX* 17.24%
V70.0	Health supervision in an adult*	75	XXXXXXXXXXXXXXXXXXXXXXXXXXX* 11.65%
73.59	Pregnancy delivered in a woman	63	XXXXXXXXXXXXXXXXXXXXXXXXXXXX* 9.78%
493.90	Asthma	55	XXXXXXXXXXXXXXXXXXXXXXXXXX* 8.54%
47.0	Appendectomy	50	XXXXXXXXXXXXXXXXXXXXXXXXX* 7.76%
V14.0	Allergy to penicillin	47	XXXXXXXXXXXXXXXXXXXXXXXX* 7.3%
28.2	Tonsillectomy	42	XXXXXXXXXXXXXXXXXXXXXX* 6.52%
28.3	Tonsillectomy with adenoidectomy	42	XXXXXXXXXXXXXXXXXXXXXX* 6.52%
401	Hypertension	36	XXXXXXXXXXXXXXXXXXXX* 5.59%
477.9	Hay fever	34	XXXXXXXXXXXXXXXXXXX* 5.28%
525.1	Teeth missing (some or all)	27	XXXXXXXXXXXXXXXXX* 4.19%
V20.2	Health supervision in a child*	27	XXXXXXXXXXXXXXXXX* 4.19%
53.00	Hernia repair	17	XXXXXXXXXXXXXXXX* 2.64%
51.22	Gall bladder removal	14	XXXXXXXXXXXXXX* 2.17%
E932.2	Estrogen therapy	12	XXXXXXXXXXXXX* 1.86%
V15.0	Allergy	12	XXXXXXXXXXXXX* 1.86%
300.00	Anxiety	11	XXXXXXXXXXX* 1.71%
564.1	Functional bowel disease	11	XXXXXXXXXXX* 1.71%
68.4	Hysterectomy	11	XXXXXXXXXXX* 1.71%
47.1	Incidental appendectomy**	10	XXXXXXXXXX* 1.55%
311	Depression	10	XXXXXXXXXX* 1.55%
706.1	Acne	9	XXXXXXXXX* 1.4%
491.2	Chronic obstructive pulmonary	9	XXXXXXXXX* 1.4%

* General physical, request for immunization, etc.

** Appendix was removed during the course of another abdominal operation.

Figure 1: A typical printout from Dr McNeill's personal computer. The chart displays the medical problems of 391 patients. The identification code is a standard set of numbers used by doctors, called ICD. ICD-9-CM stands for "International Classification of Diseases, *ninth edition, clinical modification." The meaning of each code is listed under "medical problem." The series of Xs to the right is a "histogram," or visual representation of the data. Such a chart can be prepared quickly by the computer, and gives the doctor a quick overview of the health of his patients as a group. Changes in the health of the community can be detected quickly.*

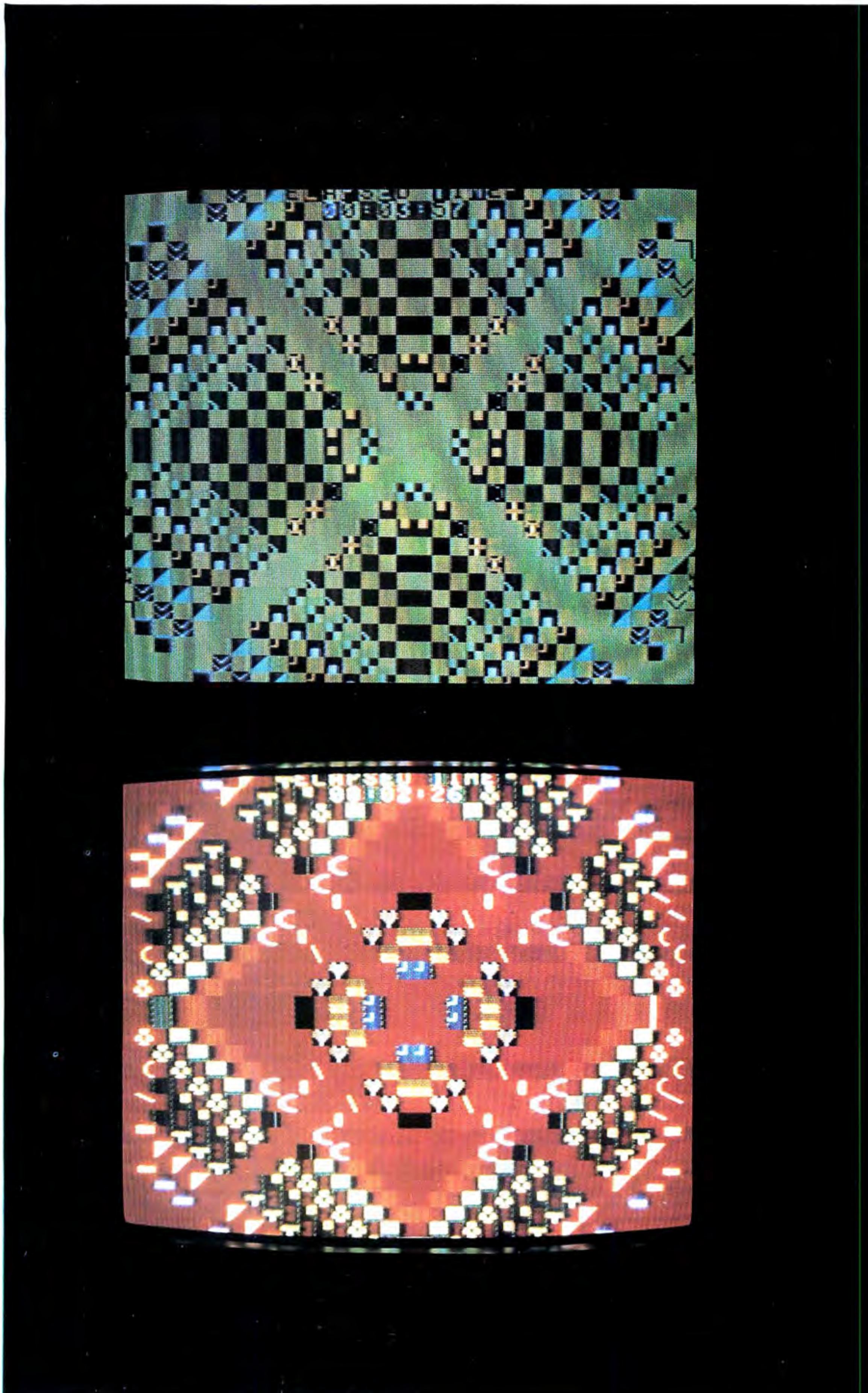


Photo 1: Color graphics on the C4P MF. These two photos, produced by a kaleidoscope program supplied with the system, show the detail and colors possible using the C4P. Notice the colored characters (pale color on darker colored background) and the "inverted" colored characters (dark color on pale color background). The message at the top of the screen giving elapsed time is constantly updated by the system's time-of-day clock.

The Ohio Scientific C4P MF --

A Computer for the Real World

by Gregg Williams

Photos by Ed Crabtree

To date, microcomputers have had little to do with the outside world. They talk mostly to cassette recorders, to disks sometimes, and to printers hardly ever. In fact, the only way they talk to *you* is through the blacks and whites (and sometimes colors) of a television set, where their impression on the outside world vanishes as soon as you turn the power off.

Electronics hobbyists building custom accessories, or suppliers of accessories for the more popular microcomputers, have designed extensions to their computers — remote security monitoring, automatic telephone dialing, even voice output. However, in the first case, not everybody wants to build custom equipment; and in the second case, there is no guarantee that extensions from different manufacturers will work together. Until recently, no one had introduced a personal computer that could interact with the outside world. Ohio Scientific is one of the first companies to address this market. Among other things, one of their new computers can detect a window being opened, dial the police, and verbally report a burglary.

The Ohio Scientific C4P MF (shown in photo 2) is an improved model based on their highly successful Challenger 2-4P computer. The additions and changes made to create the C4P signal a commitment by Ohio Scientific to capture the leadership in an industry that has recently become more professional and more competitive. In early 1977, Ohio Scientific offered one of the first BASIC language packages written by Microsoft, the company that has since written the BASIC for all of the major microcomputers. Now Ohio Scientific is attempting to expand the idea of what a "general-purpose" microcomputer should do by enabling it to interact with and control just about everything electrical in the average household.

Photo 2: The Ohio Scientific C4P MF personal computer system. Daniel Endy of the Computer Shop of Boston looks at the C4P graphics documentation while the color display shows the menu of the color demonstration disk supplied with each system. The C4P MF includes the keyboard and processor unit (left bottom, supplied with 24 K bytes of programmable memory), a single 5-inch floppy-disk drive (left top), cables, and documentation. A color monitor (or, equivalently, a color television plus a radio frequency (RF) modulator) must be added to complete the system.

The C4P MF has a number of features that make it an attractive machine. Most obvious are its use of color, its graphics (which lend themselves to arcade-like games), and its floppy-disk drive (which is used for quick access to both data and programs). Not so obvious features include its extremely fast BASIC language interpreter; a large variety of game, appliance control and security attachment options; and the particular hardware/software combination that allows the computer to interact with and initiate action through these devices. For example, the appliance control and security options allow the user to turn on all the lights in the house if an intruder is detected. Through use of the computer's

---

Gregg Williams is an editor of BYTE magazine.

design and a software feature called RTMON (real-time monitor), the computer can assume these "watchdog" tasks while it is being used for the usual tasks of a personal computer.

Hardware Features of the C4P MF

The use of color in the C4P MF is simply an extension of the video display method used in the earlier Ohio Scientific computers. The video display is given as thirty-two lines of sixty-four characters each, which can be changed to thirty-two lines of thirty-two double-width characters (used often in computer games) with a simple command. The display is *memory-mapped*; this means that the contents of a certain area of memory determines what is displayed on the screen.

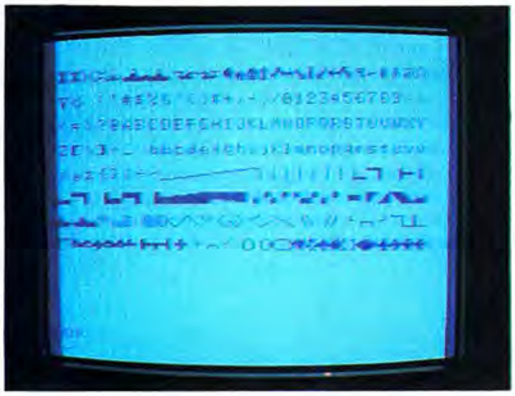


Photo 3: Graphic characters on the C4P MF. All characters on the C4P are made from an 8 by 8 dot matrix. Here they are displayed with no spacing between characters in a line and one line of space between lines. Notice the uppercase and lowercase letters and the two USS Enterprises *(on line 1). Sixteen characters not shown here include gun turrets and tanks.*

Memory-mapping, which is the method of video display used in most personal computers, allows the computer great control over the contents of the video display. It is most often used for animation—that is, to cause the illusion of movement. For example, to make a character move across the screen, first find the numeric value that causes the character to appear; then write that value to and erase it from certain memory locations in the proper sequence, and the character will appear to move.

In the C4P, two memory locations (two bytes) correspond to each character position on the video display; one location tells which character is to be displayed (out of a possible total of 256), the other tells which color is to be used and whether the character itself or the background is to be colored. A close inspection of the kaleidoscope-like picture in photo 1 will reveal colored characters on pastel backgrounds, as well as pale-colored characters on darker colored backgrounds. The C4P is capable of displaying eight colors (pastel color on darker background of same hue) and eight "inverted colors" (dark color on pastel background) for a total of sixteen color possibilities.

Memory-mapping allows the computer great control over the contents of the video display. It is most often used for animation — that is to cause the illusion of movement.

The C4P can be used with either a color monitor (which gives good color and detail, but is very expensive) or with a standard color television set. In this second case, however, a radio frequency (RF) modulator must be used between the C4P and the color television. The RF modulator changes the direct color signal to the kind of signal a color television expects to receive. The sharpness of the resulting picture may be affected by the quality of both the RF modulator and the color television. The photographs shown in this article were made using an ATV Research Microverter (cost, $35, not included with the purchase of a C4P) and a Sony Trinitron KV-1216 color television set (also not included).

Allied closely with the color capability of the C4P is its graphics capability. Since a byte of memory can hold 256 different values, one byte of the screen display memory can display one of 256 different characters. Only about one hundred characters are used to display the usual characters available on a personal computer (numbers, punctuation, and uppercase and lowercase letters), which leaves roughly 150 graphic characters available for display. These include tanks, airplanes, boats, even a *USS Enterprise* figure, as well as various abstract symbols that can be used to build larger figures; these characters are shown in compressed form in photo 3. The kaleidoscope of photo 1 was made by combining the color and graphics options of the C4P.

One of the most important features of the C4P MF is the 5-inch floppy-disk drive included with the system. Ohio Scientific has been steadily improving the operating system associated with its floppy-disk-based systems. (The *operating system* is the set of machine-language programs that

coordinate the interaction between the disk and the various user languages and utilities—BASIC, assembly language, and other functions available to the system.)

The versatile new operating system, OS-65D version 3.1, allows the disk to do a number of sophisticated things. For example, it can cause a certain program to start running (without having to type in LOAD and RUN commands) as soon as the disk is inserted; this is useful when programs are to be run by children or by adults who are not familiar with the machine. The computer can also run any sequence of BASIC and machine-language programs automatically, thus allowing the user to run a program that is too big to fit into memory.

Several other hardware features of the C4P MF deserve mention. First, all C4P MFs are shipped with a system clock that is twice as fast as the one in earlier Ohio Scientific machines and in the non-disk C4P (2 MHz clock versus 1 MHz); this means that most programs, both machine language and BASIC, will run twice as fast. This is especially important in some of the arcade-type graphics games written in BASIC, which are much improved by the increase in speed.

Second, all C4Ps include both a digital-to-analog (D/A) converter and a tone generator. Both of these can be used (only one at a time, however) to generate music, sounds, and noises that can be used to enhance any game or program. An added advantage of these options over the same options in other computers is that the signal is not preprocessed for you; you can connect the output of either of these devices to a tape recorder or a high fidelity amplifier. The optional Ohio Scientific color television/monitor has a direct input for both video and audio (which has a nice option that allows you to adjust the loudness of the sounds produced via the television's volume control); otherwise, the audio out signal is strong enough that, when using an unmodified color television and an RF modulator, a small speaker can be directly driven from the audio output jack.

Third, the C4P has two hardware timers called the *time-of-day clock* and the *countdown timer*. The time-of-day clock, once set, keeps track of the time in hours, minutes, and seconds, just as a watch would. It can be referenced by either a BASIC or a machine-language program. The countdown timer, given a time in hours, minutes, and seconds, counts down until the given time has expired, then signals the computer by means of an internal flag. Both of these timers can be used in association with the "watchdog" mode of the computer to provide services at a set time.

Fourth, the C4P keyboard is of a nonstandard design unique to Ohio Scientific products. Although this *polled keyboard* acts exactly as a standard keyboard would in BASIC, internally it is constructed quite differently. The result of this difference is that you can poll (or question) the keyboard to see which key or keys are being pressed at any given time. (Compare this to conventional keyboards, which can accept only one key at a time as being the one currently pressed.) This allows the C4P keyboard to detect a multiple number of keys being pressed simultaneously (which is useful in two-player games). It can also be used to detect key presses without using the INPUT statement (which displays the keys pressed on the video display); this can be used, say, to enter passwords. The nature of the polled keyboard allows every key on the board to automatically repeat if it is held down for more than half a second. The keyboard also allows the entry of both uppercase and lowercase letters, although the function of the shift keys is nonstandard and a bit awkward.

Photo 4: View of the C4P MF rear panel and circuit boards. The rear panel contains many plugs that allow the C4P to be connected with various peripherals. Included are connections for a modem, a printer, the BSR X-10 appliance control system, a digital-to-analog converter, video and audio outputs, joysticks, keypads, and other attachments. The circuit boards are, from top to bottom, the 505 processor board, the 527 24 K static memory board, and the 542 revision B-1 video board. The circuit board in the background is the underside of the keyboard, and the blue cable connects to the floppy-disk enclosure (not visible).

Finally, every option available to the

C4P has a plug associated with it on the rear panel of the C4P (see photo 4). This has two important implications: first, that the machine has been *designed* to use these options; and second, that these options can be plugged in *as is* without the necessity of modification by the dealer or the factory.

Photo 5: The C4P MF keyboard and optional peripherals. On the right is the command console of the BSR X-10 appliance control system. With this option, the computer can control any appliance in the home via a signal sent over the house wiring. The joysticks, center, enhance arcade-like video games. These and other options plug directly into the rear panel of the C4P.

Optional Hardware for the C4P MF

Perhaps the most innovative and useful option available for the C4P is the home security option AC-12P, priced at $175. This adapts the new X-10 remote control system from BSR for direct computer control. Basically, the X-10 system superimposes high-frequency signals along the AC wiring in your home; these signals control appliance, lamp, and wall-switch control modules (available at about $15 each). The control modules cause appliances to be turned on and off and lights to be turned on, turned off, brightened, and dimmed.

The X-10 system is very usable (for example, any control module can be overridden, allowing appliances and lights to be turned on and off manually), and the price of the system and the individual modules is reasonable. The C4P is connected directly to the X-10 command console by means of one of the plugs on the computer's rear panel.

Along with the hardware of the AC-12P (which includes the command console, two lamp modules, and two appliance modules), comes a modified operating system called OS-65D version 3.1 HC. This allows you to more easily turn appliances on and off from within a BASIC program. Also, the appliance control system is designed so that it is a part of the "watchdog" mode to be discussed later.

Just as the AC-12P option allows information originating in the computer to flow to outside devices, the AC-17P home security option allows information originating in the outside world to flow to the computer. The devices included in this option are a smoke and fire detector (sensing both smoke and heat), two window switches (that detect the window being opened), and a door switch, all of which report via battery-powered transmitters to a command console that, again, plugs into the rear panel of the C4P. Like the appliance control option, this option is also integrated into the "watchdog" mode of the computer, supplying the computer with input that may cause it to initiate some predefined action.

One option of interest to most hobbyists is the pair of joysticks available from Ohio Scientific for $39 (shown in photo 5). Joysticks are used to achieve two-dimensional control of an object on the video screen (usually during an arcade-type game). Each unit includes the joystick, a "fire" button, and a connector that plugs into the rear panel of the C4P. Since the joystick and button status are accessible from either a BASIC or a machine-language program, existing games can easily be changed to use joysticks where appropriate.

Several other devices complete the list of hardware options for the C4P. Dual keypads (small keyboards containing only numbers and a few other symbols) can be useful in both business and gaming applications. Both a printer accepting the standard RS-232C input at a rate of 300 to 1200 bits per second (bps) and a standard 300 bps modem can be attached to the C4P MF via the rear panel (however, only one may be in use at a given time). The printer is used to get paper listings of programs or data, while the modem is used for communicating with other computers over standard phone lines. As before, both options can easily be used by either a BASIC or a machine-language program.

Software Features of the C4P MF

The major language included with the C4P MF is a 9-digit BASIC written for Ohio Scientific by Microsoft (they have written similar BASICs for other major computers like the Apple II (Applesoft), Radio Shack TRS-80 Level II, Commodore

PET, Exidy Sorcerer, and Heathkit H8). It is a full BASIC that compares favorably with other microcomputer BASICs; and, running on the 2 MHz clock of the C4P MF, it is one of the fastest microcomputer BASICs available.

The C4P MF also includes Ohio Scientific's 6502 assembler and extended machine-language monitor on diskette. Both of these programs are used to develop programs in 6502 assembly language; they can be converted (or *assembled*) to machine-language programs, which will execute directly on the C4P's 6502 microprocessor. Both the assembler and the extended machine-language monitor are for use by the advanced hobbyist who wants to use the computer's machine language. Actually, since a machine-language program can be called by a BASIC program, it is possible to write hybrid programs that use BASIC for ease of programming and 6502 machine language for speed.

The RTMON program (which stands for "real-time monitor") is easily the most important piece of software given with the C4P MF because it allows the computer to coordinate the operation of "real world" devices simultaneously with the normal use of the computer. Every 40 ms, the computer checks the countdown timers and the "real world" devices that you have specified to be watched; it does this without disturbing whatever program is running on the computer at the time. Only when an active timer reaches zero or a specified device switches on to its predefined active state (for example, when the smoke detector detects an abnormal increase in temperature), does the computer interrupt the currently running program and begin executing the program on disk that has the name RTMON; this is the "watchdog" mode mentioned earlier.

Depending on the options you had programmed (which you specify in BASIC in the RTMON program), the computer could turn on an alarm, activate a device that calls the police department and plays a prerecorded message, or blink all the lights in the house. Unfortunately, the interrupted program is lost (according to the *C4P Operator's Manual*), but Ohio Scientific may come up with a solution to keep this from happening, which would make the RTMON even more usable.

The time-of-day clock, mentioned in the hardware section, allows a computer program to execute certain actions at a given time. For example, using the C4P MF with the appliance control option, it is simple to write a program that will turn appliances and lights on and off in a random but plausible manner while you are away on vacation.

Optional Software for the C4P MF

Among the list of utility programs listed for the C4P MF are a set of assembly-language utilities (65D Aux. 1), a set of color graphics routines (Graphics 1), a home control program for the appliance control and home security options (Home Control 2), and a set of music production subroutines using the digital-to-analog output (DAC Routines 1). Although I have not seen any of these packages, my recommendation (which applies to this and any other software product) is to see it demonstrated before buying if at all possible. Otherwise, a look at the documentation provided with the program is usually a good indicator of the program's quality.

Documentation

Photo 6 shows the documentation included with the system. The most impressive and complete manual is the *C4P Operator's Manual*, which documents rather well the operation of all of the components of the system. Sometimes the explanation of a given component is terse, leaving the reader to infer the implications of the material. However, this is a common shortcoming of almost all computer documentation. Without this documentation, the C4P MF would be a much less impressive system.

Four other manuals, one book, and a set of schematics complete the system documentation. The four manuals docu-

Photo 6: Documentation for the C4P MF. The documentation includes, top to bottom and left to right, manuals for the OS65D-V3.1 disk operating system, the extended machine-language monitor, the 6502 assembler and the C4P, the book BASIC and the Personal Computer *by Dwyer and Critchfield, and the Microsoft disk BASIC manual. Not shown are a looseleaf binder and a set of hardware schematics.*

ment the operating system (OS65D version 3.1), the 6502 assembler, the extended machine-language monitor, and the Microsoft disk BASIC.

Ohio Scientific also includes the book *BASIC and the Personal Computer* by Thomas A Dwyer and Margot Critchfield. This is a good tutorial book on teaching BASIC to the beginning hobbyist, and its inclusion in the C4P MF documentation will be greatly appreciated by anyone who buys the system, especially those people new to computer programming.

The Ohio Scientific Philosophy

The C4P MF, like other computers in the Ohio Scientific line, gives a lot of computing power for the money. However, this value is not achieved without a certain amount of trade-off, and an understanding of the trade-off involved is very important to the potential Ohio Scientific customer.

Photo 7: Example of a C4P MF program using video color. The purpose of this short program is to change the color of the screen and print the number of the current color. The ends of the FOR loop (lines 50 and 200) cause all sixteen colors to be displayed. Line 100 brings a machine-language program that does the color changing from disk into memory. Line 110 tells the computer which color to use next by POKEing the color number, QX, into memory location 20433, then causes the machine-language program to execute using the USR function in BASIC. Line 150 displays the current color number onscreen. Line 160 causes a wait of about fifteen seconds before the color changes again. The color showing when this photo was taken is color 10, purple.

There are two reasons for the comparatively low price of Ohio Scientific systems. The first reason is their decision to build new products around the hardware and software of their existing products. This enables them to produce a new machine for a lower price simply because money does not have to be spent to redesign the new product from the ground up. This is the case with the C4P, which is an extension of the well-designed Challenger 2-4P.

The second reason for Ohio Scientific's low prices is the more important one. The extensions to the basic computer (appliance control, color, sound, etc) are designed so that they exist in hardware only (as opposed to a design using both simpler hardware and some software to achieve the same result). A given extension is designed so that its behavior is controlled by the numeric value in a single byte or several bytes of the computer's memory—that is to say that these extensions are themselves memory-mapped, just as the contents and color of the video display are.

But it is at this point that Ohio Scientific made a decision that decreased the price of their product at the expense of its convenience of use. The decision was made *not* to modify existing software (BASIC, for example) for the benefit of the new hardware extensions. This has certain advantages, not the least of which is that virtually the same software is used by *all* the Ohio Scientific computers of the same model. (This is a problem with the Apple II, the PET, and the TRS-80.) The disadvantage is that the resulting software is not as "neat" and as easy to use as the software of other systems, a fact that unnecessarily scares away some people new to computing.

Here is an example. In the Apple II, the following program will set the eighth character on the third row to the color red:

```
100  GR
110  COLOR=1
120  PLOT 8,3
```

The first line sets the Apple II to its color-using (graphics) mode. The second line sets the current color to red (which we looked up in a table and found to be 1). The third line plots the color red onto the eighth column, third row.

The C4P would do the same thing as follows:

```
50   DEF FNP(COL,ROW)=53247+
     COL+ (ROW-1)*64
100  POINT=FNP(8,3)
110  POKE POINT,32
120  POKE POINT+4096,2
```

The first line is a function that will be used throughout the program; it converts a column and row number to the corresponding memory location of that point on the screen (remember that both character and color are memory-mapped). The second line calculates that number for column 8, row 3 (the number is actually 53383), and assigns it to the variable POINT. The third line causes that memory location given by POINT to be given the

value 32 (which I found to be the ASCII code for a blank, from looking it up in a table). The last line assigns a color (2 means red—ie: white character, red background). Since a blank is all background, a red point is produced.

The BASIC statement:

```
POKE M,V
```

puts the value V in memory location M. The BASIC statement:

```
V=PEEK(M)
```

finds the current value of memory location M and assigns it to the variable V. These two functions, plus lists of memory locations and values given in the documentation, are used to control every external device that connects to the C4P—joysticks, appliances, color television, and so on.

So the trade-off is value per dollar versus convenience of use. The documentation supplies the necessary information for the computer's use, so the decision to buy or not buy the C4P is strictly one of preference.

Choices Among the Ohio Scientific Family

This article has been primarily about the C4P MF, a desktop floppy-disk computer with 24 K bytes of computer memory, which sells for $1695. There are three other similar computers: the C4P with no floppy disk and 8 K bytes of memory, which sells for $698; the C8P, the same as the C4P, but in a larger case that allows expansion, for $895; and the C8P DF, similar to the C4P, but with 48 K bytes of memory and two 8-inch floppy disks, which sells for $2597. The versions without disks are slower and less flexible. Both versions of the C8P have extra room for more memory and more options; in particular, the C8P can house a universal telephone interface that can directly receive and dial phone calls and can even deliver a synthetic voice message. (However, it is very possible that the C4P can use these options sometime in the future.) Careful study of Ohio Scientific's current literature (available from them at 1333 Chillicothe Rd, Aurora OH 44202) will give you a better idea of which unit is best for you.

Conclusions

Ohio Scientific is moving in a direction that indicates better service, better products, and better software, all available at a reasonable price. But the desirability of a computer depends on many things outside the merits of the hardware and software. The following paragraphs give four points that deserve some consideration.

Ohio Scientific's current line of software is weak but should improve, given the new direction they seem to be taking. As I said before, I recommend checking out either the program or its documentation before buying.

Ohio Scientific deals with customers only through its network of dealers. Of course, the dealer's proximity to you is important. But equally important is his or her service to customers, since you will be working with the dealer for service and information about your system.

Also, very few people or companies have written software for the hobbyist Ohio Scientific computers so that, presently, you are limited to buying Ohio Scientific software or writing your own. I am, however, aware of two companies that currently supply OSI software. Mark Bass (269 Jamison Dr, Frankfort IL 60423) supplies a BASIC-and-machine-language graphics utility called GRAFAX for $11 postpaid. Aardvark Technical Services (1690 Bolton, Walled Lake MI 48088) has some nice games and utilities; their catalog, available for $1, has some interesting information on the non-disk Ohio Scientific computers. However, the disk-based systems do not have a cassette interface, and both of these suppliers (to my knowledge) sell their software in cassette form.

Finally, think before you buy. The Ohio Scientific computers are not "appliance computers"—that is, they aren't designed primarily for the plug-in-and-run type of user who doesn't want to know anything about *how* the computer works (although this would be an excellent system on which to learn about computer hardware). To get the most out of such a machine, you should be curious enough about computers to experiment with them, customize existing programs to your needs, and write programs on your own. If this describes your inclinations, the Ohio Scientific C4P MF deserves your serious consideration. ■

Our sincere thanks to the Computer Shop of Boston and Cambridge for its assistance with the photographs made for this article.

DO YOU FEEL LIKE TALKING?

*MAIL
+s ASEJALLEN

.Hello old buddy. Do you feel like talking?
.

Mail waiting
+pu
#9 ASEJALLEN 04/05/79 12:04:30 *

Sure. How are you?

+s ASEJALLEN

.Great. Tell me about yourself. Do you like skiing,
.skydiving, girls, boys...studying? What do you
.do for kicks?
.

Mail waiting
+pu

#10 ASEJALLEN 04/05/79 12:12:14 *

I do lots of groovy things. I'm not into skydiving though, are you?

+s ASEJALLEN

.I'm not into that either, skydiving I mean. I do like
.men though; do you?
.

Mail waiting
+pu
#11 ASEJALLEN 04/05/79 12:18:32 *

Shall I assume that you are not male?

+s ASEJALLEN

.Never assume, but you could be right. What year
.are you in school?
.

Mail waiting
+pu
#12 ASEJALLEN 04/05/79 12:24:10 *

I'm a freshman. What are you?

+s ASEJALLEN

.A sophomore. Hey, I have to go or I'll be late for
.anthropology class. Thanks for the chat. I'll be
.back later, OK?
.

Edward J Gauss is an Associate Professor of Computer Science and Electrical Engineering at the University of Alaska in Fairbanks. Alyce J Egan is doing postgraduate research at the University of Alaska considering the human aspects of computing.

by Edward J Gauss and Alyce J Egan

Students in remote corners of Alaska can now talk to one another by computer "mail."

Dialog out of a cheap novel perhaps? No, this is modern-day "instant" mail—computer style. Like citizens band (CB) radio, it has a style all its own. Being fairly unusual in structure and application, the MAIL System at the University of Alaska is revealing new insights about people's behavior. Through it individuals can communicate with their peers or with others inside the University Computer Network — anonymously, if desired. A state which relies heavily on radio and telephone communications to shorten the great physical distances between communities for business and personal dealings, Alaska is also seeing the development of communication patterns of another nature.

Advanced technology has created a means by which a person may seek, maintain, and terminate communication with others while remaining unseen, nameless and physically silent. Given an opportunity for anonymity, the personality an individual projects may be quite different from that used in a face-to-face dialog. Several factors are involved in this. Shy or inarticulate people can gain a measure of self-confidence when their "true" selves are not exposed. Thus they become able to express themselves more openly with little fear of failure.

Computerized electronic mail also provides an opportunity for the individual to project an image that may be partially or totally invented. This is true in citizens band radio. There, each individual has a code name of his or her own invention that may represent a fantasied self-image. A sense of fraternity exists and each person demonstrates his or her knowledge of the ritual in order to prove membership. There are no binding obligations, and each person can leave the fraternity simply by throwing a switch. The technology protects each member from face-to-face encounters, allowing a different CB personality to be projected.

Similar communication has developed within the University of Alaska Computer Network. There are three factors which have made this possible:

- *The MAIL System.* This is a program system adapted for the University of Alaska network by Ron Hansen. It enables the computer to receive and dispatch messages which are typed in from any one of the computer terminals scattered about the state.

- *Liberal Use Policy.* The university has a policy of "free" use of the computer by its students and staff.

- *Configuration.* As many as eighty terminals may be simultaneously connected to the computer. They are distributed widely over the many campuses of the University of Alaska and are located for the most part in small clusters to create a feeling of privacy.

We have observed several different communication modes at the University of Alaska and have come up with a simple classification system for them:

Illustration by Robert Tinney

1. Can you top this?
2. Hello there!
3. Charlie McCarthy.
4. Hear me!

There are no formal instructions or rules for any of these communication techniques, but informal patterns do tend to exist, creating loosely defined yet recognizable structures of their own.

Such diversions or "games" may even be beneficial mechanisms in the learning process. All of these begin when a person is sitting at the relative privacy of a computer terminal. A terminal is similar to a typewriter, however, with the keyboard used for typing information into a computer. It may have paper which prints everything that has been typed or the results of programs run, or even a screen similar to a television set to display information. Each terminal is connected directly or by telephone to the central computer housed in the Fairbanks campus. While the main function of the computer is for university-related business activities, it is widely used by the students and university personnel for academic and personal use. Wastebasket studies have shown the importance of the illusion of isolation. There is far more activity in small rooms with only two terminals than at the central location where faculty and staff are present.

To begin, the person "logs on" and then types MAIL, which connects him or her into the system. He or she will be told by the computer if messages are present and can then browse through them at his or her leisure. Messages can be saved, deleted, or even sent on to someone else. They can also be separated and saved in different files, much as one might separate letters into file baskets for incoming, outgoing, business, or personal inspection at a later time. Thus a business message may be dealt with and removed, whereas a love letter could be reprinted any number of times or saved and reproduced again at some future date.

"Can you top this?" is a communication "game" played with a single partner. The purpose of the game is to continue the communication and to insure that you make the last reply. Whatever the partner says must invoke a reply from you. Some players can run three or more partners at once, all with different and independent communication dialogs. The only limit is one's ability to say something clever or unique. Such bantering may lead to more serious dialog or continue off and on in this vein for weeks. Players of "Can you top this?" will inconvenience themselves in order to send the last message. This may make them late for an examination. The game is an activity that meets Wright's definition of friendship (see the references listed at the end of this article). He requires that the partners willingly accept inconvenience to continue mutual activity. If we accept Wright's view that "friendship depends for its existence upon the way the persons involved *see* or interpret *each other*," we may say that computer technology enables the establishment of friendships, however tenuous or contrived they may be. There is always the potential for rewarding communication to bond the friendship together.

```
**Mail waiting**
+pu
#23 ASEJALLEN 05/08/79 10:32:41 *

I'll be coming to Fairbanks next week. Can I meet
you somewhere?


+s ASEJALLEN

.Sure friend. Give me a call when you get to town.
.You still have my telephone number don't you?
.
```

In "Hello there!" the person is seeking new communication partners. A computer command enables the person to learn the code names of everyone currently working on the computer. A message is simply sent to the user code to see if a reply can be obtained, as in "Hello old buddy. Do you feel like talking?" The first letter of the code (A) identifies the campus and the second letter (S) identifies the classification of the person. Thus a student, code S, may identify a faculty member, code F, and avoid him or her for this game. An Anchorage student, code A, may seek replies from distant Juneau students, code J, or Fairbanks students, code F. A graduate student, code T, may be approached with curiosity, and a staff member, code Y, may be approached, but only after normal working hours. A few rounds of "Can you top this?" result until it is established that each wishes to play. There is no cue to the gender of the partner. This often interests the male college student, and he may enter the game hoping to locate a new girl friend, albeit one that is several hundred miles away. So the game variation of gender identification often develops. If a question is too direct, the partner may convert the game to "Can you top this?" and evade the answer for days. If the question is too obscure, an even more obscure reply may be invoked. This is a rather successful means of establishing new communication partners, particularly on the intracampus level.

Charlie McCarthy was Edgar Bergen's famous ventriloquist's dummy who made statements which were uninhibited by social convention. He could say things that his master could not. The computer is used by some people in this way. "Charlie McCarthy" is played by people who know each other.

The computer provides an isolation that permits statements of a personal nature that could not be made in a face-to-face encounter. Games of this type have been observed between two people who were both located in the same room. Both were fully aware of the physical presence of the other and that they could turn and face each other at any time. A duality has been observed. A mechanical malfunction may require one person to leave his terminal and directly help the other. This causes a shift in projection to the face-to-face mode. When the trouble is resolved the person returns to his or her terminal and shifts projection back to the computer mode. Thus the game can be pursued through the *implied* anonymity of the computer. Questions and statements may get quite personal in nature, be cutting or extremely intimate. Yet, as with Charlie McCarthy and Edgar Bergen, almost any subject is fair game to be pursued to any depth.

"Hear me" is a rather one-sided game and the one with perhaps the greatest distortion of self. An audience of strangers is selected. The MAIL System enables a single message to be sent to this entire audience by means of a single command. "Hear me" players may have trouble in dealing with individuals on a direct basis. The computer provides a method for an individual to force a large audience to receive his communication. The potential for swaying others to one's own views is great. The audience can be varied by campus, rank, or other means, yet still remain anonymous. The reaction of the audience is varied. The computer is not immune from the junk mail problem, and some people are hostile to such unwanted messages. Some welcome communication from a safely distant stranger, but one who is acceptable because he knows the ritual of the computer. A computer pen pal may result.

The rather unsanitary method of wastebasket inspection indicates that an active communicator will process ten to twenty messages in a session. This may take several hours and be interspersed with some efforts at schoolwork. It appears that the schoolwork is prolonged so that the player will be present for all of the possible replies that might be coming.

It appears then that the computer network has become a breeding ground for person-to-person communication not before realized. Through it, an individual may find a new friend or a sounding board for new ideas. According to Wright, friendships take some time to develop as the persons involved develop a voluntary interdependence on each other. Early friendships tend to expose each partner to facets of the other not revealed in more formal interaction. The process of developing a potential friendship which, under conventional means, may take some time to establish and play itself out, may be speeded up, altered, or tested along the way, through the computer. A strong friendship can endure much

strain and still remain intact.

If a weak friendship encounters strain, however, the partners usually abandon the relationship before the strain becomes intense. Such abandonment generally is gradual as partners no longer find any ego-support value or stimulation in the relationship. The computer frees both parties of mutual responsibilities or deep emotional commitment. Simply by pushing a button, friendships may be dissolved with lightning speed yet have little overt emotional effect on either partner. And as both parties are intrinsically aware of the possible personality contrivance of the partner, theatrical tendencies may flourish. But just as Charlie McCarthy could be put back into his box at the end of the show, so can a computer partner be turned off to be dealt with again when convenient.

Even pen pals must put more effort into a relationship. A pen pal at least knows the name and address of his or her partner, and must not only write words to him or her, but must find an envelope, get a stamp, and see that the letter gets mailed. The computer provides for instantaneous sending of a thought to another and having it instantly received. Spontaneity becomes the basis for quick wit. Possible actions may be tried with countless variations on a countless number of people with very little effort.

Nearly half a million messages were sent in 1978. These were not uniformly distributed over the 5000 active users. Certainly the use of the computer for communication without commitment is an attractive idea for many students and staff of the University of Alaska. The amount of time and inconvenience suffered is substantial. Between 10% and 30% of the occupancy of computer terminals is by persons who are involved in communication-related use.

The computer MAIL System must be fulfilling some strong need for these students. Greeley states that friendship is "... rhythm, an alternation of mood, sentiment, ... from one partner to another, a flow of signal and countersignal that enables each partner to be more fully himself and to facilitate the enrichment of the other." It does not matter to what degree the friendship ensues. The relationship itself is a learning experience, and that, after all, is what colleges are designed to provide.

It has been said that "reality, however utopian ..., is something from which people see the need for taking rather frequent holidays." The computer network can provide a means for escaping reality, establishing needed friendships, yet causing little harm and perhaps much benefit to those with access to the knowledge of the MAIL System.

Mail waiting

+pu

#34 ASEJALLEN 02/01/79 08:42:19 *

Good morning friend. Have you got time to talk?

+s ASEJALLEN

. Sure friend. What's the weather like in Anchorage?

. It's -30 here today . . .

References

Berne, E, "Transactional Analysis: A New and Effective Method of Group Therapy," *American Journal of Psychotherapy*, December 1958, page 735.

Greeley, A M, *The Friendship Game*, Doubleday and Company Inc, Garden City NY, 1970, page 79.

Hansen, R, "The MAIL System," computer program, University of Alaska Computer Network, 1976.

Powell, J T, and D Ary, "Communication Without Commitment," *Journal of Communication*, 27, 3, 1977, page 118.

UACN Newsletter, University of Alaska Computer Network, March 1979.

Wright, P H, "Toward a Theory of Friendship Based upon a Conception of Self," *Human Communication Research*, 4, 3, 1978, page 196.

U.S. POSTAL SERVICE

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

(Required by 39 U.S.C. 3685)

1. TITLE OF PUBLICATION: onComputing
A. PUBLICATION NO.:
2. DATE OF FILING: Oct. 1, 1979

3. FREQUENCY OF ISSUE: Quarterly
A. NO. OF ISSUES PUBLISHED ANNUALLY: 4
B. ANNUAL SUBSCRIPTION PRICE: $8.50

4. LOCATION OF KNOWN OFFICE OF PUBLICATION *(Street, City, County, State and ZIP Code) (Not printers)*: 70 Main St. Peterborough, N.H. 03458

5. LOCATION OF THE HEADQUARTERS OR GENERAL BUSINESS OFFICES OF THE PUBLISHERS *(Not printers)*: 1221 Avenue of the Americas New York, N.Y. 10020

6. NAMES AND COMPLETE ADDRESSES OF PUBLISHER, EDITOR, AND MANAGING EDITOR

PUBLISHER *(Name and Address)*: Virginia Londoner Vale St. Peterborough, N.H. 03458

EDITOR *(Name and Address)*: Carl T. Helmers Jr. Carriage Hill Rd. Hancock, N.H. 03449

MANAGING EDITOR *(Name and Address)*: Christopher P. Morgan Hancock Rd. Dublin, N.H. 03444

7. OWNER *(If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be giving. If the publication is published by a nonprofit organization, its name and address must be stated.)*

NAME	ADDRESS
PLEASE SEE ATTACHED SHEET *	

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES *(If there are none, so state)*

NAME	ADDRESS
NONE	

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES *(Section 132.122, PSM)*
The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes *(Check one)*
NOT APPLICABLE
☐ HAVE NOT CHANGED DURING PRECEDING 12 MONTHS ☐ HAVE CHANGED DURING PRECEDING 12 MONTHS *(If changed, publisher must submit explanation of change with this statement.)*

10. EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES PRINTED *(Net Press Run)*	77,018	77,233
B. PAID CIRCULATION 1. SALES THROUGH DEALERS AND CARRIERS, STREET VENDORS AND COUNTER SALES	15,458	15,840
2. MAIL SUBSCRIPTIONS	30,393	32,566
C. TOTAL PAID CIRCULATION *(Sum of 10B1 and 10B2)*	45,851	48,406
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	5,991	906
E. TOTAL DISTRIBUTION *(Sum of C and D)*	51,841	49,312
F. COPIES NOT DISTRIBUTED 1. OFFICE USE, LEFT OVER, UNACCOUNTED, SPOILED AFTER PRINTING	24,833	27,921
2. RETURNS FROM NEWS AGENTS	344	none as yet
G. TOTAL *(Sum of E, F1 and 2—should equal net press run shown in A)*	77,018	77,233

11. I certify that the statements made by me above are correct and complete. SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER

12. FOR COMPLETION BY PUBLISHERS MAILING AT THE REGULAR RATES *(Section 132.121, Postal Service Manual)*

39 U. S. C. 3626 provides in pertinent part: "No person who would have been entitled to mail matter under former section 4359 of this title shall mail such matter at the rates provided under this subsection unless he files annually with the Postal Service a written request for permission to mail matter at such rates."

In accordance with the provisions of this statute, I hereby request permission to mail the publication named in Item 1 at the phased postage rates presently authorized by 39 U. S. C. 3626.

SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER

PS Form 3526 *(Page 1)* Aug. 1978 (See instructions on reverse)

STATEMENT OF OWNERSHIP, MANAGEMENT, AND CIRCULATION

*7) The owner is McGraw-Hill Inc., 1221 Avenue of the Americas, New York, New York 10020. Stockholders holding 1% or more of stock are: Donald C. McGraw Jr.; Harold W. McGraw Jr.; John L. McGraw; William H. McGraw; June M. McBroom; Elizabeth McGraw Webster; all of 1221 Avenue of the Americas, New York, New York 10020; Occi and Company, c/o Occidental Life of California, P.O. Box 3635, Terminal Annex, Los Angeles, California 90054.

Put the BYTE on the IRS with Aardvark™

In an era when computers are an integral part of business and entertainment, and computer software is more and more sophisticated, Aardvark™ is yet another breed in advanced computer software. Aardvark's™ Micro Tax series is a true user-oriented Federal Income Tax package. Specifically developed by qualified tax professionals for use on personal home computers, this low-cost, time-saving Aardvark™ Micro Tax package accurately and efficiently computes your federal income tax liability. Aardvark™ will display and fill in facsimile Form 1040 and related schedules and, when connected to a printer by means of a standard interface, will print out these facsimile forms automatically. Each program is designed to use the appropriate tax table or rate schedule. The Aardvark™ package includes an indexed instruction manual and input forms for easy input of tax data. To see how Aardvark™ can be tax-deductible, check the instructions in the binder of the manual. You can also use the binder to store pertinent 1979 tax records. Cure your headaches this tax season, order today . . . and put the BYTE on the IRS with Aardvark™!

MICRO TAX I — $25
Form 1040
Schedule A
Schedule B
Schedule TC
(will not calculate Income Averaging, Max Tax, or Alternative Minimum Tax)

MICRO TAX II — $35
Form 1040
Schedule A
Schedule B
Schedule G (Inc. Avg.)
Schedule TC
Form 4625 (Min. Tax)
Form 4726 (Max. Tax)
Form 6251 (Alt. Min. Tax)

MICRO TAX III — $50
Micro Tax II plus
Schedule C
Schedule D
Schedule E
Schedule SE
Form 2119 - Sale of Personal Residence
Disk Version — $60

MICRO TAX IV — $100
Expanded disk version of Micro Tax III
Printed forms can be filed with the IRS

When Ordering Specify:
TRS-80 16K Level II Basic
Apple II 32K with Applesoft Basic
Apple II 16K Micro with Applesoft ROM Card
TI 99/4 with drive cassette
PET 16K

Aardvark Software

MICRO TAX V — $150
Expanded disk version of Micro Tax III plus Form 2210
Provides for hash totals and batch processing
Printed forms can be filed with IRS

For more information write:
P.O. Box 26505
Milwaukee, WI 53213

WE HONOR master charge THE INTERBANK CARD — **24 hrs.** Orders only call TOLL FREE 1-800-558-8570. **7 days** — VISA

In Wisconsin 1-414-289-9988

Circle 120 on inquiry card.

A Beginner's Guide to Programming Languages

by Steve A Hughes

Many people hesitate to enter home computing because they know little about programming and are worried that it may be difficult to learn. They are told that it is no worse than learning a foreign language—which immediately brings to mind the horrors of irregular verbs. Helpful computing friends carry on high-speed conversations full of terms like "one-pass assembler," making them wonder if it would not be easier to learn Basque than programming. The possibility of ever learning programming starts to look more and more remote. The purpose of this article is to shed some light on the subject and reassure those who have been terrified by well-meaning friends that the art of programming is really not so mysterious after all. Let us begin by disposing of the idea that learning programming languages resembles learning foreign languages.

No Irregular Verbs

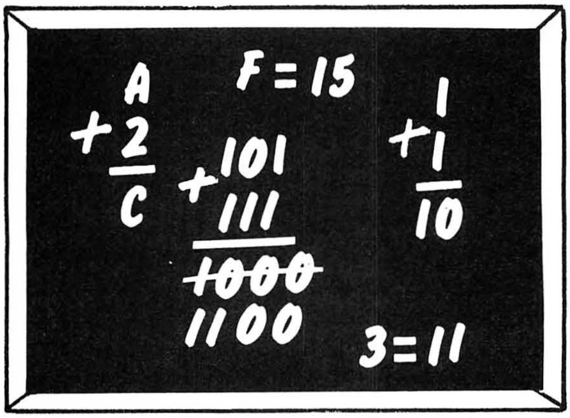
The most commonly taught foreign languages (eg: French, German, Spanish, and so on) involve the student with an unfamiliar vocabulary; endings to specify case, tense, and gender; and the need to be able to speak the language coherently and understand it when it is spoken by others. Of these concerns, only the need to learn a different vocabulary is applicable to programming (and in many higher-level languages, the vocabulary to be learned consists almost entirely of English words or abbreviations for English words).

It is a better analogy to compare instructing a computer to do something with giving directions to a small child whose grasp of English is limited. Suppose you want the child to put a toy in the toy box. You must first be sure it understands which toy you mean and where to find it; second, direct it to pick up the toy; and third, instruct it to put the toy in the toy box. Telling a computer to store an item of data in memory is a very similar process. You must indicate where the piece of data is to be found (an input port, for instance) and that it is the piece of data you mean; then you must direct the computer to accept the data; finally, you must instruct it to store the data in a specified location in its memory. If you are dealing with an older child, whose command of English is better, the shorter instruction "Put the toys away" will suffice. Similarly, a computer which has been programmed to convert briefer instructions into the form it needs may be able to act on the command "DATA 16, 27, 93, 12, 48, 62" by storing the data specified (the numbers in the example) in memory.

Programming, then, might be described as suiting your conversation to your hearers. You would not use ten-syllable words if you were trying to communicate with a child who could not understand them. In the same way, you have to present instructions to the computer in a form it can use. So the crucial question is, "What can the computer understand?"

Machine Language

A computer's fundamental language is its *machine language*, which it usually shares only with other computers of the same model. Since everything is represented inside the computer in patterns of 1s and 0s, the instructions in its machine language are represented this way also. These *operations codes* (or op codes, for short) are the computer's words of one syllable only. They require no further translation and can be executed promptly. Thus, 10110100 might mean *jump*—that is, go to a specified next instruction instead of the one that happens to occur next.



Because codes using only 0s and 1s (*binary* codes) are hard for people to remember, two other number systems are frequently used to represent op codes. They are *octal* (using the eight symbols 0 thru 7) and *hexadecimal* (using the sixteen symbols 0 thru 9 and A thru F). They are easy to remember and easy to convert to the binary code the computer needs. To find the octal equivalent of a binary number, replace each group of three binary digits with its octal equivalent. You start at the binary point (the binary equivalent of the decimal point) and work to the left to derive the integer part, or to the right to derive the noninteger part, of the number. Hexadecimal (or hex, as it is sometimes called) conversions work exactly the same way, except that each group of four binary digits is replaced with its hexadecimal equivalent. Figure 1 shows this process.

Yet even octal or hexadecimal codes can be difficult to learn or remember, and programming in machine language can have other disadvantages. Consequently, *assembly languages* come into the picture.

Assembly Language

In assembly language, the numbers that represent commands in machine language are replaced by easier-to-remember mnemonics. For example, the command to store an item, A, in memory might be simply STOR A in assembly language. By permitting the use of variable names and other symbols, assembly language greatly simplifies the task of *debugging* (ie: correcting) programs. In machine language, a jump instruction would have to specify exactly where to jump—for instance, to jump twenty-five instructions. If, in debugging the program, you added or deleted instructions anywhere in those twenty-five instructions, the jump command—and probably other instructions as well—would have to be changed to reflect the fact that the instruction you wanted to jump to would no longer be twenty-five instructions away. In assembly language, however, the ability to handle symbols lets you say, for instance, "jump to B" instead of giving a precise specification. The computer, when it reaches that instruction, can go unerringly to B regardless of how many or how few instructions appear between the jump command and B.

Some assembly languages also offer another useful feature: the ability to use *macroinstructions*. A macroinstruction is a single mnemonic code, user defined which represents several machine language commands performed in a set order. Using macroinstructions, you can reduce the need to write out the same sequence of instructions repeatedly, thereby saving time and reducing the risk of careless error.

The use of assembly language on a computer is made possible by a program already stored in the computer's memory, called an *assembler*. This program takes the assembly-language code you enter (the *source code*) and converts it into machine-language code that the computer can use (the *object code*). To do this, it must prepare a *symbol table* which defines the variable names and other symbols that occur in the program; convert the commands into machine codes; resolve any macroinstructions into their component machine codes; and organize the whole into a machine-language program the computer can execute. Sophisticated assemblers can do all this in a single pass through the source code (hence the term one-pass assembler), but many assemblers require more than one pass. A three-pass assembler, for example, might use one pass to prepare the symbol table; a second to convert the assembly-language instructions into machine codes and to organize the program in a form convenient to the computer; and a third pass to create a human readable listing of the program. Then a *loader* program, which may or may not be linked to the assembler, must load the resulting object code into the computer. Only then can the program be run.

Unfortunately, there are two severe disadvantages to relying wholly on assembly language. The first disadvantage is that assembly language, like machine language, is *machine dependent*; that is, an assembly-language program can rarely be run on any other model of computer than the one for which it was written. This means that every program must be rewritten for each computer on which it is to be run. The second disadvan-

Figure 1a: Conversion from binary to octal and from octal to binary.

Octal Symbols	Binary Equivalents
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111

Binary Number:	10	110	100	.01
Octal Equivalent:	2	6	4	.2

Octal Number:	7	3	5	1.	4
Binary Equivalent:	111	011	101	001.	100

Figure 1b: Conversion from binary to hexadecimal and from hexadecimal to binary.

Hexadecimal Symbols	Binary Equivalents
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

Binary Number:	111	0100	1011	0110	.1111
Hexadecimal Equivalent:	7	4	B	6	F

Hexadecimal Number:	D	2	C	9	1	E
Binary Equivalent:	1101	0010	1100	1001	.0001	1110

> *. . . no programming language has been developed so far to let programmers tell the computer what they want in their own words.*

tage is that most people find assembly language nearly as difficult to learn as machine language. So, although machine and assembly languages are still heavily used in some applications, most programming is currently done in what is called a *higher-level language*.

Higher-Level Languages

The earliest higher-level languages were developed to meet specific needs. For example, *FOR*mula *TRAN*slator (FORTRAN) was developed for mathematical calculations and *CO*mmon *B*usiness *O*riented *L*anguage (COBOL) for business needs. Other languages for specific applications have been developed, such as *LIS*t-*P*rocessing Interactive Language (LISP) and Stri*N*g-*O*riented Sym*BO*lic *L*anguage (SNOBOL) for functions involving character strings. However, the current trend is toward more general-purpose languages which can be used effectively for many different sorts of applications. *ALGO*rithmic *L*anguage (ALGOL), the most popular language in Europe, has features of a general-purpose language, and Pascal and IBM's *P*rogramming *L*anguage I (PL/I) are general-purpose languages which are gaining large followings in this country. Pascal language is named after Blaise Pascal (1623-1662), the French mathematician and philosopher.

Higher-level languages are essentially of two types, *compiler languages* and *interpreter languages*, distinguished by how the source code (ie: the original form of the program) is converted into a form the computer can use. This is not to say that any particular language, like *B*eginner's *A*ll-purpose *S*ymbolic *I*nstruction *C*ode (BASIC), can be implemented only in one of these ways, for the source code will look the same regardless of which approach is taken to rendering it machine usable. It is true, however, that each language usually is more commonly implemented one way than the other. Thus, FORTRAN is usually implemented as a compiler language, while BASIC is most often an interpreter language.

Compiler languages utilize a program called a *compiler* to convert the source code into a form the computer can use. It prepares a complete object code form of the program before executing it, sometimes taking advantage of the source code's structure to make the object program more efficient. Usually, more than one machine-language instruction is required for each statement in the source code.

An interpreter language, on the other hand, utilizes a program called an *interpreter* to permit the computer to execute each statement in the source code as it occurs. The interpreter scans the source-program command determining what operations are required, performs these operations, and then moves on to the next statement in the program. Naturally, this approach prevents any preliminary optimization of the code, and frequently requires that the same statement in the source code be interpreted several times in the course of running the program. The result is that interpreter languages run more slowly than compiler languages. Similarly, since both types require considerably more conversion to reach a form the computer can use than does assembly language, a program in a higher-level language can rarely achieve anything like the speed of a comparable assembly-language program.

In applications where this inevitable loss of speed is unimportant, the use of higher-level languages brings many advantages. Since a single higher-level language command can replace many instructions in a lower-level language, the programmer's task is eased. As with the use of macroinstructions in assembly-language programming, this reduction in the number of steps the programmer must write out reduces the risk of careless error. More important to the beginning programmer, the resemblance many higher-level languages bear to English in their statement structure and syntax tends to make them easier to learn. The strongest argument for the use of higher-level languages, however, is that they are *machine independent*. A program written in COBOL can be run with little or no modification on any computer for which COBOL is available. This means that many programs need only be written once, instead of needing to be rewritten for each different computer on which they must be run. This factor alone makes the use of higher-level languages desirable, even where ease of learning and coding are unimportant.

Although programs exist which let the computer give coherent responses to ordinary conversational sentences, no programming language has been developed so far to let programmers tell the computer what they want in their own words. It is still necessary to formulate instructions in a precise fashion, as defined by the programming language in use, in order for them to be converted appropriately into machine codes and executed. But with the advent of higher-level languages, the world of computers became open to anyone willing to devote a little time to learning a programming language.

The Right Language for You

When you start shopping around for a higher-level language to use on your home system, several questions arise. Does the language you are considering meet your applications needs? For that matter, is it available for your computer? How much memory does it occupy? Do you have enough memory to be able to store the compiler or interpreter and still have room for the programs

you want to write? Does using the language introduce any special needs, such as a special character set, and if so, are you equipped to handle them? How easy is the language to learn and use? How much software has already been written in the language that you might be able to use? Each of these considerations can be important in your decision.

The most critical factor in selecting a language is your proposed application. Although most languages can be adequate for a reasonable number of purposes, a specialized application can exceed the capabilities of a more or less general language. By the same token, a specialized language might be almost impossible to use for a purpose other than the one for which it was designed. Thus, FORTRAN, while a very good language overall, would be an extremely poor choice for word processing applications; while LISP, designed to handle character strings, might be unsatisfactory for an application dealing with scientific calculations. Your particular application, then, can make some languages wholly undesirable while making another language or group of languages preferable or necessary.

Moreover, the decision to use a special-purpose language can have far-reaching side effects. Few languages are available for more than a small number of personal computers. In consequence, if you require an uncommon language, you may also need the computer system for which it is available. This possibility makes it wise to look into your language needs before investing in computing equipment, unless you are prepared to write any specialized language you may need for your computer system.

... look into your language needs before investing in computer equipment, unless you are prepared to write any specialized language for your system.

Your hardware (ie: equipment) can have another effect on your freedom to use languages. The compiler or interpreter program which permits you to use a higher-level language must be stored in the computer's memory before you can use programs in that language. Therefore, if the language you want to use fills 4 K (4096) bytes of memory with its compiler or interpreter, it cannot be used on a system which has only 4 K bytes of memory. No memory would be left for storing and running your program. Fortunately, some languages are available in versions requiring different amounts of memory, but the smaller versions, of necessity, have fewer functions, slower operation, or similar limitations. The need for a full implementation of a common language or for a specialized language, consequently, can require an addition to your computer's memory.

The ease with which the language can be learned and used is the next concern. If you do not already know the language you propose to use, having good textbooks accessible can be extremely helpful. If the language has special requirements, such as the special character set required by *A Programming Language* (APL), it is essential to know that you can satisfy them. It is always worthwhile to look for any possible difficulty with a language in advance. Doing so, you can avoid the danger of finding that you have paid a high price for something you cannot use without a further, and possibly larger, investment.

A final consideration is the availability of prewritten programs in the language. If you can get satisfactory programs for many of your tasks simply by transcribing them from magazines or by ordering them from software houses, you are that much better off. In BASIC, for example, some applications are so well covered that the amount of programming you must do is reduced to practically nothing.

The question then is what higher languages are available?

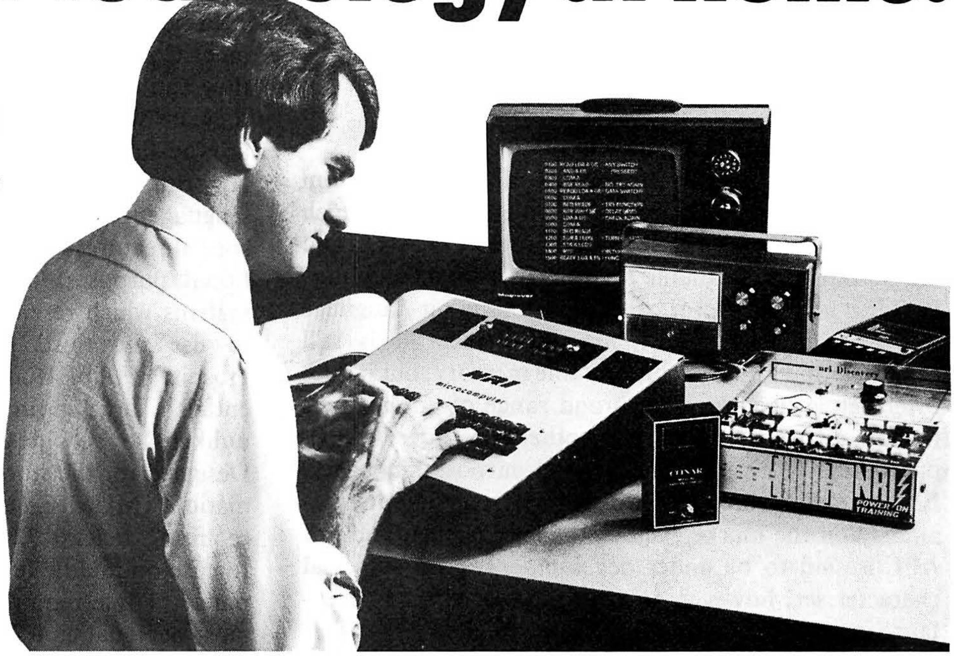
BASIC

As the dominant language in personal computing, BASIC has advantages in any application that does not require specialized capabilities. It is available, usually in several versions, for practically every personal computer on the market. The larger versions (12 K or 16 K bytes of memory) routinely offer more decimal-place accuracy in calculations (important in some applications), increased array-handling capabilities, and more or less extensive functions for dealing with character strings. The smaller versions (up to 2 K bytes) provide a form of the language which is satisfactory for a reasonable range of applications without requiring extensive memory. Since BASIC was developed as a language for teaching programming, it is one of the easiest languages to learn and use. Many textbooks are available to help the beginner, and countless BASIC programs have been published for a wide range of applications.

Since the versions of BASIC for personal computers do not necessarily conform to an *American National Standards Institute* (ANSI) standard, they are less uniform in features and format than are languages which customarily adhere to such standards. However, this lack of uniformity rarely creates serious problems. Although programs written in a more complex BASIC may be unusable with a smaller BASIC, most programs in one BASIC can be run using another BASIC of equal or greater complexity with few or no modifications.

Availability and ease of use have been the keys to BASIC's success. Until another language can offer a similarly extensive range of versions and prewritten programs for computer systems, it will probably remain the most important personal computing language.

Build your own microcomputer as you learn computer technology at home.



New from NRI! The First Interdisciplinary Home Study Course Ever Offered

As the microprocessor revolutionizes the computer world and microcomputers appear almost everywhere, NRI brings you a new, convenient, and effective way to keep up with this expanding technology. It's NRI's courses in Microcomputers and Microprocessors, created and designed exclusively for learning at home in your spare time.

Designed for the New Breed of Computer Technician

It's no longer enough to be just a programmer or technician. With microcomputers moving into the fabric of our lives as low-cost, easily available tools for business and home, both the programmer and technician must become total professionals. With practical knowledge of hardware, the programmer can design simpler, more effective programs. And with advanced programming skills, the technician can test and debug systems quickly and easily. The NRI course gives you simultaneous training in both skills...makes you one of this rare new breed.

Build Microcomputer, Test Instruments

NRI goes far beyond book learning to give you practical, "hands-on" experience. As you learn, you actually assemble NRI's designed-for-learning microcomputer. It performs like the finest of its kind, and features both assembly and basic language capabilities.

Every assembly step's a learning step. Using the NRI Discovery Lab® plus the NRI transistorized volt-ohm meter and CMOS digital frequency counter you also build, you perform meaningful experiments throughout your course...trace circuitry, interface components, introduce and correct problems, design your own programs, and more.

The Proven Way to Learn at Home

You don't have to worry with travel, classes, or time lost from work when you learn the NRI way. As they have for more than 60 years of teaching technical subjects, NRI brings the material to you. You study in your spare time, at your convenience, using "bite-size" lessons that program material into logical segments for easier assimilation. You perform experiments and build equipment using kits we supply. And your personal NRI instructor is always available for consultation should you have questions or problems. Over a million students have already shown the effectiveness of NRI training.

Choice of Courses

Several courses are available, depending upon your needs and background. NRI's Master Course in Microcomputers and Microprocessors starts with the fundamentals, explores basic electronics and digital theory, the total computer world, and the microcomputer. The Advanced Course, for students already versed in digital electronics, concentrates on software and the world of the microprocessor and microcomputer. In both courses, you build all instruments and your own computer.

Send for Free Catalog... No Salesman Will Call

Get the details on these exciting new courses in NRI's free, 100-page catalog. Shows all kits and equipment, lesson outlines, and full information, including facts on other electronics courses. Mail the coupon today and we'll rush your catalog. No salesman will ever call. Keep up with the latest technology as you learn on your own computer. If coupon has been removed, write to NRI Schools, Computer Department, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Amateur Radio • Basic and Advanced
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating Including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 177-020

Circle 130 on inquiry card.

Many programming problems cease to be problems when you know a few of the terms and understand that it is all a question of trying to tell the computer, as clearly as possible, what you want to do.

General-Purpose Languages

There are challengers however—most notably from the ranks of the all-purpose languages. PL/I, which combines features of COBOL, FORTRAN, and ALGOL, is now available for some personal computers in a less complex form called *P*rogramming *L*anguage For *M*icroprocessors (PL/M). At present it is still expensive and requires considerable memory, but it does offer better coverage of a broad range of applications than many other languages. Another contender is APL, one of the more powerful general-purpose languages. EMPL, a version of APL for 8080-based computers, is already on the market, and a fuller implementation of APL is said to be under development. APL's special character set, however, is apt to slow down its acceptance in personal-computing circles.

The most important alternative to appear from among the general-purpose languages is Pascal, which has been the subject of considerable discussion by computer enthusiasts in recent months. Pascal, an extremely powerful language that combines the bit-manipulation capabilities of assembly language with the range of operations usually associated only with higher-level languages, is having a noticeable impact on professional computing, despite its comparatively recent development. Its importance in personal computing is increasing as versions of it are announced for more and more systems. Although all the implementations of Pascal to date require mass storage (which may be a deterrent to those whose needs can be met by a less expensive arrangement), Pascal is rapidly becoming a major personal-computing language. Whether it will replace BASIC as the dominant language remains to be seen.

Old Standbys

Two of the older computing languages, widely used in professional computing, are also appearing for personal systems. COBOL, the prevalent business language, known for its verbosity and for easing the generation of reports, is now available; and FORTRAN, the most common scientific language, is available for the larger personal computers which can execute operating systems like Digital Research's CP/M product. FORTRAN IV, conforming to the ANSI standard FORTRAN X3.9-1966, is offered for Z80 computers. Although ill-suited to games applications where extensive formatted output is foreseen, the increasing availability of FORTRAN has been a real boon to personal-computer users who are concerned with mathematical and scientific applications—and to the tens of thousands of programmers who already know FORTRAN and who do not want to learn a different language.

Programmers who are familiar with Digital Equipment Corporation (DEC) equipment will find also that they can continue to use three of DEC's languages if they are willing to select computers for which versions are available. *F*ormulating *O*n-line *C*alculations In *A*lgorithmic *L*anguage (FOCAL, DEC's BASIC-like language) and DECAL (DEC's language offering special features to ease the use of computers for instructional purposes) can now be obtained for personal systems which resemble DEC equipment. The decision to use the third language, *M*assachusetts General Hospital *U*tility *M*ulti*P*rogramming *S*ystem (MUMPS), entails a large enough equipment expenditure that it is unlikely to become common in personal computing. Designed for the storage and retrieval of medical information, MUMPS is a data-base-oriented language requiring at least 32 K bytes of memory and intended for use on middle-level to high-level PDP-11 minicomputers. These requirements make it far too costly for most personal-computer users, but there are hopes for versions of MUMPS for less expensive systems.

Specialized Languages

The only specialized application that is well provided with higher-level languages for personal computers is word processing. LISP (an interactive list processing language that offers ease of learning, simple syntax, and the ability to use programmer-defined functions) is available for several computer systems. *STRU*ctured *BA*SIC *L*anguage (STRUBAL), which is available for the 6800, combines extensive functions for handling character strings with the same sort of readability-by-the-uninitiated that characterizes BASIC programs. SNOBOL, a text-editing language suitable for tasks such as generating concordances, has also become available for a few systems. While a certain amount of care in selecting equipment will be necessitated by a desire to use a string-handling language, the field here is far less narrow than in some other applications areas.

One of the most widely used specialized languages is PILOT. The name stands for *P*rogrammed *I*nquiry, *L*earning *O*r *T*eaching, and the language is designed for use in instruction by computer. As such, it simplifies the task of accepting and responding to conversational input, while being sufficiently easy to learn that an instructor who lacks programming experience, but wishes to use the computer as a teaching tool, can readily learn it. The versions available for personal computers sometimes limit computational capabilities severely in order to keep the memory requirements

small, but this is rarely a serious weakness. In the applications for which PILOT is most desirable, extensive computations are seldom important.

In another applications area altogether is CMS-2Z (one of the less well-known languages). An enhanced version of the United States Navy's CMS-2 compiler, designed for command and control applications, it is also suitable for scientific and real-time tasks. CMS-2Z has no formatted output, but in the applications for which it is intended, formatted output probably would not be necessary.

Also originally developed for process control, but equally well suited to a broad range of applications, is FORTH (not an acronym). FORTH permits the definition of new operations by the programmer and, on most systems, the use of any number base (ie: binary, octal, decimal, etc). However, many people find it difficult to learn, and program listings are difficult for the "uninitiated" to decipher. Like most languages, though, FORTH has a dedicated following. If your equipment is suitable, you might consider trying it and deciding for yourself.

Conclusion

Being able to write your own programs provides two major advantages. If you want to use your computer for an application for which no prewritten programs exist, you can write the programs you need yourself. And if you find that a prewritten program has weaknesses, you can correct them instead of being forced to live with them. Programming is not the esoteric science it sometimes appears to be. Many programming problems cease to be problems when you know a few of the terms and understand that it is all a question of trying to tell the computer, as clearly as possible, what you want to do.

If you are trying programming for the first time, I suggest you start with BASIC. Easy to learn and readily available, it will let you get comfortable with programming while providing enough capabilities to handle the most common applications. When you find yourself interested in specialized applications, you should be able to make the transition to a more powerful or more specialized language with minimal difficulty.

As you can see, quite a few higher-level languages are already available in versions for microcomputers — and the number is always increasing. In this article I have tried to provide the background you need to make sense of reports on languages and an overview of the sorts of languages that are available.

If you have not tried programming yet, why not start now? Creating good programs can be fun and the benefits of being able to provide your own programming are enormous. ■

Learning About Computers: So Easy a Child Can Do It

by Jim Heter

A twelve-year-old boy sits down with a packet of written material and a small computer, and with little or no guidance completely masters the operation of the computer in a few days. Child prodigy? Not at all. At the Delphian School in Sheridan, Oregon, this sort of thing is a routine occurrence. Such performance is, in fact, required of all students on their way to graduation.

At the Delphian School, all of the more than 300 courses offered in the kindergarten thru twelfth grade curriculum are programmed; not in the multiple branching sense of the usual "programmed learning" course, but in the sense of having checksheets which present a sequence of steps to guide the individual student through the course. The checksheet lists both material to be read and practical exercises to be done, broken down into easily digestible pieces. These must be completely assimilated, one-by-one, with all new words carefully defined, and plenty of "hands on" drills and exercises.

I have written four courses on computers and computing which are now included as part of this curriculum. Some examples of what these courses contain and how the material is presented may be of interest to personal computer owners, because it is a uniquely successful approach to self-instruction that is well adapted to the personal computer field. No lecturer or expert instructor is needed, because all of the data and the practical exercises necessary to ensure understanding are included in the material. It is helpful, of course, to have a somewhat knowledgeable person at hand who can occasionally check the student's progress and help him or her track down the source of any difficulties encountered. In the classroom this role is filled by the course supervisor, who takes the place of the teacher in a conventional classroom.

My fourth course, entitled simply *Introduction to Computers,* has not yet been printed. It is a more elementary course on computers and computing, with the purpose of familiarizing students with the uses of computers and giving them a good grounding in the operation of the simplest and least expensive electronic computers available — hand-held calculators. The course includes data sheets on Computation, Storage, and Control, the three basic uses to which computers are put. It also has a series of data sheets on the theory and operation of calculators, with exercises to help the student attain proficiency in their use. An interesting feature of this course is the use of an abacus to explain the action of both decimal and binary registers in a data sheet called *The Digital Computer.*

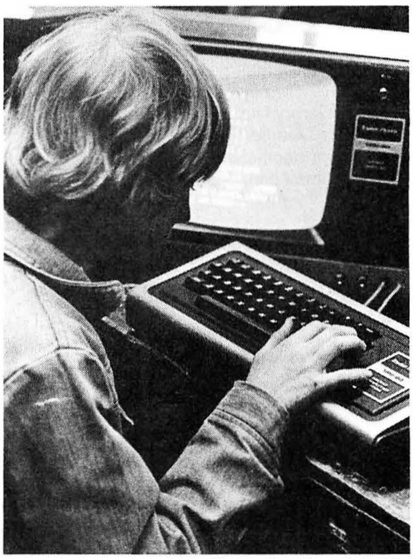


Photo by Frank Mills

An intriguing possibility with all of these courses, and indeed with most of the other courses in the Delphian curriculum, is that of programming the checksheet and all of the written material into the computer, to be called up to the display in what is automatically the correct sequence. Equipment limitations have kept us from doing this as yet (print on paper is still cheaper and more abundant than magnetic tape and video displays), but I look forward to doing something with this soon on a trial basis. Given sufficient memory access, the potential for interactively facilitating the study process, with inclusion of glossary terms and computer-monitored drills, is too attractive to ignore.

[On the following pages we present selections from some of the outstanding course materials developed by the author to introduce his students to personal computing. The courses are designed for use with the Radio Shack TRS-80 personal computer — ed]

What Is a Computer?

A computer is "a person or thing that computes." Compute means "to do by counting or by arithmetic." Usually when we say "computer" we are talking about an electronic computer. Electronic means "making use of the behavior of small particles called electrons." Many things can be done by carefully following a few simple rules about the behavior of electrons. Making electronic computers is one of these things.

An electronic computer is a sort of artificial mind. It is a little harder to use than your own mind, because it only understands special languages made of numbers, but it is a little easier to understand than a mind because it is much simpler. A computer can count, do arithmetic, and remember a lot of numbers. If you or someone else tells the computer that a certain number stands for a certain word or letter (whoever does this is called a programmer), the computer can remember this information.

You usually "talk" to a computer by typing on something that looks like a typewriter keyboard. The computer usually "talks" to you by writing on something that looks like a TV screen.

If a programmer has told the computer that certain numbers stand for certain words, then the computer can "talk" to you in words that you can read on its screen.

What Is a Computer Game?

A computer game is one that is played by using an electronic computer. There must be a "program" for the computer, which is a long list of numbers that tell the computer what the rules of the game are, and how you want the game displayed on its screen.

There are things you can tell the computer to do that it will understand. These are called your "moves," or "turns," or "entries"; they are what you can decide to do as you play the game.

The rules of the game tell you what moves you are allowed to make, and tell the computer what it has to do each time you make a move.

And, of course, a computer game has purposes. These are the things that you have to get done in order to win the game. Part of the purpose is to get enough points to win, or to find the right answer to some question, or to find the right procedure to get the right result. Another part of the purpose is to learn *how* to win the game, and then to get good at it by practice. Yet another purpose is to give you a chance to apply things that you had learned before.

Computer games are different from other games because the computer takes care of remembering the rules and remembering the score, and it won't let you break the rules or change them in the middle of the game. Even if you are playing the game by yourself, you can get the feeling of being in a real situation where you aren't the only one who cares what is happening.

Computer games can be very complicated, since the computer automatically keeps track of everything and doesn't make mistakes. This helps to make the games more interesting and more fun to play.

More About Games

After you have run a computer and played a computer game, you should know a little more about games and why computer games are good for more than just having fun.

In a way, life is a game. Many people have trouble thinking about life that way, because they get so involved in it that they think it is much too serious to be a game. But life, like other games, consists of freedoms, barriers, and purposes. So when you play any game, you can say you are practicing for the game of life.

Computer games make especially good practice for the game of life. The computer can keep track of things that are quite involved and complicated, yet can keep them from becoming too confusing. The game of life, when you think about it, is very complicated, mostly because there are so many players.

The nice thing about practicing for life by playing games on the computer is that you *know* that they are really games, and you don't have to be too serious about them. Someday when you have to apply what you learned with a computer to real life, you will already know how well you can do it.

The TRS-80 Microcomputer

Section 1

What is the TRS-80? TRS-80 means *T*he *R*adio Shack-80; 80 is from Z80, the microcomputer circuit made by the Zilog company which Radio Shack used in designing the TRS-80 system.

You don't need to know too much about the TRS-80 in order to turn it on and run it. Like most computers, the hardest thing about using it is to write good programs (instructions that tell the computer what to do with information). Once that has been done, the computer can basically run itself, and a person can use the program without knowing what is really going on inside of the machine. That is why the computer seems almost alive to some people; it does complicated things for them in mysterious ways that they don't understand. You can plan on solving all of these

mysteries for yourself as you study more about computers. For now, the main things to know are those which will allow you to use the machine.

The TRS-80 is called a "Microcomputer System." Micro means very small. Compared to some computers the TRS-80 is very small indeed, but because of recent advances in modern electronics technology, it is also very good. New technology has made it possible for very small computers to be actually better than the very big computers used to be. (Of course *modern* big computers are even better still.)

The reason it is called a Microcomputer *System* is because the TV-type video display, the data-entry keyboard, and the program record and playback cassette deck are not part of the computer itself, which is a very small electronic device hidden inside the keyboard console. They are what are known as computer peripherals. "Peripheral" means "away from the central part." They are needed so that you can understand what the computer is doing, or so you can tell it what to do. The computer is the central part. The computer along with the peripherals make up the computer system.

Section 2: How to Run the TRS-80 Computer

A. Connecting It Up

1. Plug in the video display.
2. Plug in the power supply.
3. Connect gray cable from video display to video jack on back of keyboard.
4. Connect gray cable from power supply to power jack on back of keyboard. (First make very sure keyboard power switch is off.)
5. Plug in the cassette recorder.
6. Connect the cable with three plugs to the recorder:
   Black plug to EAR jack
   Larger gray plug to AUX jack
   Smaller gray plug to REM jack
7. Connect the other end to the TAPE jack on the back of the keyboard.
8. Plug the dummy plug into the MIC jack on the recorder.

B. Turning It On

1. Turn on the video display by pressing the POWER button.
2. Turn on the TRS-80 keyboard by pressing the POWER button on the back.
   The red light to the right of the keyboard should light up.
   The display screen should show READY.
   Adjust C (Contrast) and B (Brightness) controls on front of display for sharpest image. Background should be gray and letters white. Do not set brightness too high.
3. If display does not show READY, turn keyboard POWER switch off and on again.
4. To make sure everything is right, type the letters P.M. and press the ENTER key. If everything is okay, the display will show "P.M. 15871". If it doesn't, go back to Step 3. (NOTE: Don't forget the periods after P and M.)
5. There is a reset button inside a door on the back of the keyboard. Press this any time the TRS-80 doesn't do what it should. It will make the computer go back to the beginning and start over.

C. Loading the Program

1. Put the program cassette in the machine.
2. Be sure that the tape is fully rewound and all plugs are in place.
3. Set the volume control between 7 and 8.
4. Push the PLAY button down so that it locks.
5. Type the letters NEW, then press the ENTER key. This will clear out any existing program.
6. Type the command CLOAD and the ENTER key. This will start the recorder, and the program will begin loading.
   The display will show an asterisk (*) when loading starts, and a second * will blink on and off as loading continues.
   When the loading is completed, the display will show the word READY.
7. Press the STOP button on the recorder.

D. Running the Program

1. When the program has been loaded, type RUN, then press the ENTER key.
   The program will begin, and from here you can follow the directions given for the program itself. If the program doesn't start the way it should, you can go back to *Loading the Program* and try loading it again. If it still doesn't work, use the "Reset" button, then reload it. If that doesn't work, go back to *Turning It On*, and recheck all of the steps there. After that, try another program, or call your supervisor.
2. When you are through using the system, turn off the power on the keyboard and display. Then you can unplug everything if you wish.

Glossary of Computer Terms

Cable: An electrical wire or bundle of wires used to connect two parts of the system together. It carries electrical power or electrical signals.

Cassette: The magnetic tape recording package used to store a computer program when the computer is turned off or used for something else.

Display: The TV-type screen that is used to show the computer output data.

Enter: A key on the keyboard that tells the computer that the data that has been typed should be accepted and acted upon.

Jack: A connector on a part of the system where a cable can be hooked up.

Keyboard: The typewriter-like panel used to manually feed data to the computer.

Load: Put a program into the computer, especially a program stored on a cassette.

Plug: The connector on a cable that goes to a jack on a part of the system.

Power Supply: The electrical circuit that changes power from a wall socket to the correct level for the system.

Program: A list of instruction steps, written in a special simple language that the computer can understand, that must be fed into the computer so that the computer knows what to do.

Run: Start the computer going through the steps in the program and doing what it says to do.

P.M.: Stands for *Print Memory*. Tells the computer to check how much of its memory is unused and display the answer on the screen. ■

Note: Selections from curriculum materials are copyrighted by the Delphian Foundation and are used with permission.

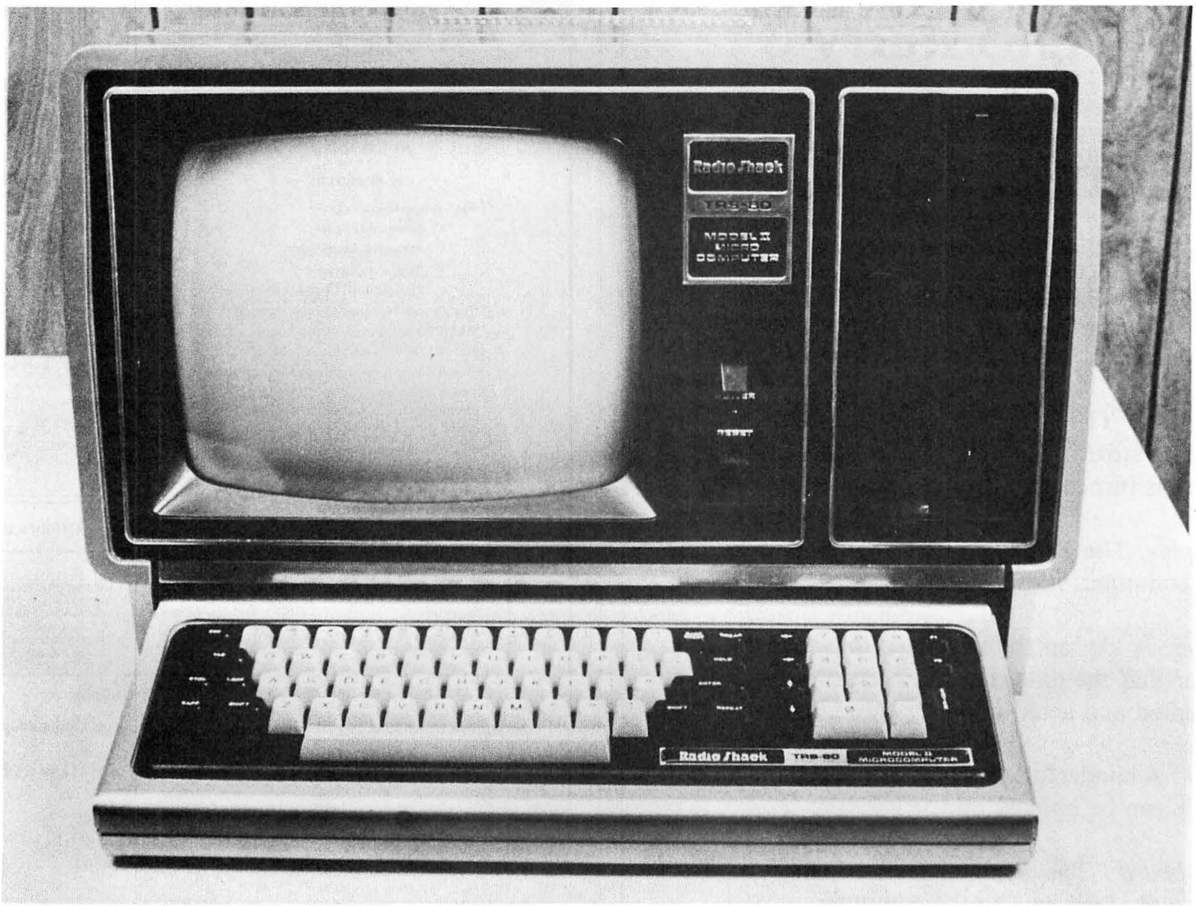


photo by Ed Crabtree

The Tandy-Radio Shack TRS-80 Model II computer system. Shown here is the basic $3450 unit with 32 K bytes of memory, Z80A processor, and single 8-inch floppy-disk drive. The unit can be expanded by adding up to three additional 8-inch drives. The TRS-80 Model II is aimed primarily at the business market. An impressive selection of business software is available, and the exhaustive owner's manual is well-written for the nontechnical user.

Radio Shack's TRS-80 Model II Computer:

An Evaluation

by Bob Magruder and Dave Barstow

After seeing the Tandy-Radio Shack TRS-80 Model II small business computer system demonstrated at several personal computer and small business shows, we were anxious to get a machine off of the assembly line to find out if all of the good things we had heard about the equipment were true.

The TRS-80 Model II is available in its most basic form for $3450 (shown in photo). It features a Z80A processor, 32 K bytes of programmable memory, a 12-inch video display, keyboard, and a single 8-inch floppy-disk drive. A 32 K byte memory add-on brings the price up to $3899. Different versions of the Model II are available, including the top-of-the-line $8737 system with 64 K bytes of memory, three external 8-inch floppy-disk drives, line printer, line printer stand, and desk.

We were able to set up and use the machine for ten days; we set it up in our computer room next to a

TRS-80 Model I equipped with full communications capabilities. Ken Jansen of Jemco Sales in Madison Connecticut, has developed several business programs on a TRS-80 Model I computer that he has used for the past several years. We assisted him with the transfer of data and programs from the Model I to the Model II. The program of highest priority for Jemco was a mailing list program with approximately 1500 names in assorted categories.

The Model II system is packed in two cartons: a small one containing the keyboard and the owner's manual, and a large one containing the main unit. No great difficulty should be encountered with the small carton, but we were advised that the large carton can present difficulty. The carton is designed in such a way that it is possible to drop the computer while unpacking it, if the carton is placed right side up and unpacked on a table. We had no trouble unpacking the machine from the floor with the carton inverted. We understand that Tandy-Radio Shack is recalling and repacking in a redesigned container those machines presently in the warehouses. However, machines presently in dealers' hands will probably not be repacked in the new container.

Setting Up the System

Setting up the system was straightforward. A terminating plug must be installed in the disk expansion port at the rear of the computer (see instruction manual) unless the auxiliary disk-drive unit is connected. The terminator is supplied with the computer. A single cable from the computer must be plugged into the keyboard, and the power cord must be plugged into the power connection on the back panel of the computer and connected to a grounded electrical outlet.

Appearance, Mechanical Design, and Keyboard Layout

Both the appearance and functional design of the TRS-80 Model II are very good. The mechanical design of the equipment appears to be sturdy. The cabinet material and color are similar to the Model I. An immediately visible improvement to the mechanical design is the elimination of six power cords and interconnecting cables. The two-piece design of the cabinet and keyboard is neat and professional. The separate, movable keyboard makes the system easy to use. The only cable visible from the operator's station is the connection between the keyboard and the computer. All other cable connections are on the rear apron of the system. The front panel contains a power on/off switch and reset switch, a single vertically mounted 8-inch floppy-disk drive (supplied with the system), and a video display screen.

Both appearance and functional design of TRS-80 Model II are very good

The keyboard layout and mechanical design are much improved over the Model I keyboard. The feel of the keyboard is as good or better than many terminals available today. It contains all of the common control keys such as Escape, Control, Break, Shift, Tab, and Back Space, as well as some uncommon ones. The additional control keys are a Shift-Lock (with indicator light), which provides all of the uppercase letters, the Caps key (with indicator light), which provides an uppercase alphabet but also provides lowercase numeric and control keys, a Repeat key that allows any character to be repeated, and a Hold key that causes a pause in program execution until it is pressed again. The Hold key allows the programmer to halt a program listing for examination on the screen.

Also included are two user-definable function keys, F1 and F2. F1 is presently assigned to the direct-statement Edit call, but may be reassigned. The direct-statement Edit is also called by typing control-O. There are four cursor control keys and a full numeric keypad with its own Enter control key. The keyboard is very close to a standard typewriter layout. It should provide few surprises to the experienced touch typist.

We have one negative comment about the keyboard. The BASIC language uses the colon (:), semicolon (;), and double quotation mark ('') extensively. Both the colon and semicolon are on the same key, as are the single and double quotation marks. This is distracting for a programmer who is programming in capital letters and who must shift to get the colon and double quotation mark each time a multiple statement line or print statement is entered. This caused us difficulty with syntax errors.

The 12-inch diagonal-measure video display exhibits excellent resolution and contrast in the normal white-on-black mode. Contrast diminishes greatly in the reverse mode-video (black-on-white), but resolution appears to hold up well. Diagonal retrace lines appear in the reverse mode, but these may be adjusted out of the normal mode by increasing the contrast slightly. The retrace lines should disappear if the adjustments on the video display are touched up slightly by an experienced television technician.

There are two serial input/output (I/O) ports and one parallel output port on the rear apron of the computer. To use the parallel output port with a Tandy-Radio Shack or other Centronics-compatible printer, the Model I printer cable must be detached from the printer and replaced with the Model II printer cable supplied with the system. Serial port connectors and cables are not supplied. Both serial ports must be correctly terminated with electrical resistance if either is to be used. If you want to use only one port, instructions are provided in the technical section of the opera-

ting system manual for the assembly of a terminating connector for the unused port. The manual has several pages of corrections that are inserted in the front of the book.

Be sure to make the corrections before trying to assemble the terminating connector or implementing the serial ports.

The TRSDOS Disk Operating System

It is extremely easy to start the system. When the power is turned on, the system tells you to insert a floppy disk. DO NOT turn power on or off with a disk in place in any disk drive. It is highly probable that data will be lost if the system is powered-up or powered-down with a disk in the drive.

When the operating system disk is placed in drive 0 (the drive next to the video screen), the computer automatically loads the operating system and performs a system test. *[The* operating system *is the "master" program in a computer that controls the operation of the floppy-disk system and other aspects of the system —ed]* In a multidrive system, the operating system must always be on a disk in drive 0. This means that even in a two-drive system, data disks may be copied only by using the Backup utility, not by using the Copy command from the system library. This may cause programmers some extra manipulations of the system, but should not concern the typical business user. Duplicate copies of records should be made after update in any environment, but this is especially important with business records.

If you are accustomed to 5-inch (actually 5¼-inch) floppy disks, you might find the operation of the larger 8-inch disks a little strange. The write-protect notch on an 8-inch disk is reversed (ie: an *un*-covered notch means that the contents of the disk are protected from change). Tandy-Radio Shack 8-inch disks are delivered with the notch uncovered (write-protected). It must be covered with one of the small labels provided in the box of disks before it can be used. Purchasers of single disks should ensure that a label set is included in their purchase or they will not be able to use the disks. 3M disks are provided without a notch cut. These disks may be used as they come out of the box, but the notch must be cut out with a knife or punch to protect them. This sounds like a minor problem, but at first we cut a protect notch in a 3M disk and then tried to format it. This absolutely will not work.

Once we had the TRSDOS operating system loaded we decided to make a backup system disk for self-protection. The process of making a backup disk went quite smoothly after a couple of false starts. There are parentheses ( ) , braces { } , and square brackets [ ] on the keyboard. None of the operating system commands use square brackets as far as we could tell. Some of the commands use parentheses to define command syntax or file specifications; others use the braces. The instruction manual often confuses the characters, but the machine never confuses them. Where blanks are specified for use in the command line they must be used, and where they are not specified they must not be used. Problems with a system command can often be solved by exchanging parentheses for braces, or vice versa.

After we found solutions for the notch and command syntax problems, the formatting and backup of the system disk went smoothly but slowly. Formatting the 77-track dual-density disk takes about six minutes with a full check for bad tracks. This procedure should be used on new disks.

The slow backup is caused partly by the necessity of repeatedly removing and inserting disks in a single-disk system, and also because in a single-disk-drive copy operation the system stores the information from the disk in user memory. As a result, only about eight tracks may be copied at a time. For a full disk backup, the operator may have to swap disks as many as ten times.

The excellent prompting messages and error detection make the job of formatting and backup nearly foolproof, but the possibility of operator error in confusing the disks during the backup procedure is high. Serious use of this system in the business environment will require at least one other disk drive.

System commands include those that activate the format and backup utility programs, as well as the commands listed on page 67. The system is complete and everything works very well. The entire repertoire of commands is also available from Model II BASIC, which makes the programming and operation of the system a pleasure.

Model II BASIC

Model II BASIC is an updated version of Model I Disk BASIC. The language was written for Tandy-Radio Shack by Microsoft. All of the statements available in Model I Disk BASIC, with a few exceptions, are available in Model II BASIC. The actions performed by statements also are the same.

The PEEK and POKE commands are missing from Model II BASIC. However, a statement that finds out and returns a value corresponding to the row position of the cursor and one that returns the column position of the cursor have been added. The PRINT AT statement has been changed because the ampersand (&) is now a shift-2, changing its value in the encoded form. User access to machine-language routines through a USR statement has been retained. Ten machine-language routines can be identified.

Some details concerning BASIC in the Model II have been altered from the Model I, as well as the methods of handling files and the file structure. (A short description of files is given in our small business introductory article elsewhere in this issue.)

The command that activates BASIC now assumes that you have

no files open. It is required that the number of files that will be used by BASIC programs be specified when BASIC is called from the operating system.

Files are no longer handled as blocks on records of the floppy disk. If a field statement does not use 256 bytes, the next record will start where the last record left off. Both random-access and sequential-access files are supported.

Two file types are supported by the system: fixed length and variable length. Variable-length files are not discussed in the user's manual except for a statement of how to create them by using the CREATE statement, and that they must be treated as sequential files. However, sequential files that are OPENed and CLOSEd with standard BASIC file statements from BASIC programs are *fixed-length* files, with a record length of 1, and are not variable-length files.

This caused some trouble in the conversion job in which we were involved. The needed information is hard to find in the manual.

Other than the few problems outlined above, the user's manual is excellent. It is divided into three sections: the first on operating the system, the second on characteristics of the disk operating system (DOS), and the third on Model II BASIC. It is well indexed and cross-references are supplied where one function affects another. Examples of program segments and operation are provided for each statement and command.

The manual contains over 420 pages of instructions, illustrations, and examples. In our use of the manual, we had to call the Tandy-Radio Shack Fort Worth "hot line" only once during our first week of intensive operation in order to clarify a point.

While it is not a Level I manual designed to teach BASIC, it is one of the best system documentation efforts we have seen provided with a microcomputer.

The technical section of the manual is primarily aimed at the assembly-language programmer. Many assembly-language subroutines and functions have been built into the system to facilitate programming in a lower-level, faster-operating language. No editor-assembler program set is provided with the machine, however. Our assembly-language programming is being done on the Model I, assembled on the Model I, and then transferred to the Model II via the communications subfunction UPLOAD. Debugging and revisions are then managed in machine language using the DEBUG utility program.

System Commands for the TRS-80 Model II

AGAIN — allows any command to be repeated
APPEND — adds the contents of one file to another
ATTRIB — changes the file protection attributes
AUTO — provides an automatic boot into a program
BUILD — allows the creation of chained commands and programs
CLEAR — clears user memory (above hexadecimal memory location 27FF)
CLOCK — turns on clock display (CLOCK OFF to turn off)
CLS — clears screen
COPY — copies one file to another disk (*Requires disk name assigned during* FORMAT)
CREATE — allows user to pre-allocate file space
DATE — provides day-date data
DEBUG — machine-language monitor (DEBUG OFF to turn off)
DIR — disk directory and status of files and file space
DO — used to start a command file created with BUILD
DUMP — stores a machine-language program on disk
ERROR — provides an English translation of system error codes
FORMS — sets printer parameters and control codes
FREE — provides a map of the space on a disk
I — initializes a new disk without initializing system area
KILL — deletes a file
LIB — provides a list of the system library
LIST — provides a record listing of a file
LOAD — loads a machine-language program file
PAUSE — halts execution for an operator action
PROT — changes a disk protection level (*Requires original master password assigned during* FORMAT)
PURGE — deletes all files of a specific type or all files (*Requires disk master password assigned during format and defaults to* ALL FILES *on* DRIVE 0, *including the* OPERATING SYSTEM)
RENAME — renames a file
SETCOM — sets up and enables communications channels (initialized to OFF by the system at startup time)
TIME — gets the time of day (set at system initialization)
VERIFY — read after write verify (ON to turn on, OFF to turn off)

Program Conversion from Level I to Level II

One of the selling points for the Model II is the ease with which Model I programs can be converted for use on the new system. This was also a selling point for upgrading from Level I BASIC to Level II BASIC on the Model I. As before, ease of conversion is a relative term.

Conversion software that is now in the Tandy-Radio Shack Computer Centers sometimes works and sometimes does not. The conversion programs are in machine language, so the chances of finding people in

the store to fix them are slim. Therefore, we strongly suggest that users for whom conversion is a factor in purchase should have their software converted and demonstrated by the dealer before they accept delivery of the Model II.

Several parts of TRS-80 Model I programs will not convert directly to Model II programs:

- The PRINT AT statement will not convert, but will run if retyped.
- All PEEKs and POKEs must be removed or conversion will fail.
- All USR calls must be removed and rewritten. Machine-language programs must have the sections that perform input and output operations (I/O drivers) rewritten because these are located in different places in memory in the Model II than in the Model I.

Finally, there are several versions of the conversion connection cable available, only one of which works.

These problems are not unusual with a complex piece of equipment relatively new to the field. They will be solved as more of us gain experience with the machine.

Tandy-Radio Shack Software

Tandy-Radio Shack software provided for the system is oriented to the business user. It includes the new business mailing list system and the word-processor and accounting package. As of this writing, the software has not yet been released. Reports are that the word-processing system program is a fully conformal word-processing system to be sold with a Tandy-Radio Shack "daisy-wheel" printer similar to the Diablo printer. A serial-printer driver modification for the operating system may become available.

Additional Software and the CP/M Operating System

Osborne/McGraw-Hill small business program packages are presently being converted by Bill Driscoll at Small Business Systems Corporation by time-sharing the Model I and Model II with a large computer. The integrated business software should be available by the time this issue is published.

Lifeboat Associates, 2248 Broadway, New York NY 10024, has converted the popular CP/M operating system for use with the Model II. All of the CP/M software should then be usable by the Model II. The FORTRAN, COBOL, and Pascal computer languages will also be supported under the CP/M operating system.

Recommendations

Based on our experience with the Tandy-Radio Shack Model II, we believe that this computer deserves serious consideration from the small business computer user.

One final note: in our opinion the TRS-80 Model II should be bought with 64 K installed. TRSDOS and BASIC take up 27 K, leaving only 5 K of user space in a 32 K machine. 64 K is available as an initial 32 K board plus a second 32 K board installed at a later date, using two slots in the backplane. However, when the machine is purchased as a 64 K machine, it is supplied with a 64 K memory board using only one slot in the backplane and reserving four slots for future expansion of the memory to the planned 212 K. This is especially important to CP/M operating system users.

We have found several problems with the conversion of software from the Model I to the Model II, but no significant problems in the conversion of data from one machine to another. File handling differs between these machines, however, and there may be problems that we have yet to uncover. In any event, we strongly suggest that business users who want to convert older TRS-80 Model I programs should request a demonstration before the sale is concluded. We further suggest that the converted data should be reorganized with a utility program to take advantage of the Model II's speed and power.

The Model II is well built, well designed, and priced to provide strong competition in the microcomputer marketplace well into the 1980s.

Instruction manuals and documentation provided with the computer are excellent. We have not seen any better in the industry. Manuals, however, must be read and used to be effective.

Support for the hardware and programs through computer stores with trained personnel will be vital to the success of the Model II. The machine is neither more nor less intelligent than an IBM System/370. It will require strong support from Tandy-Radio Shack to be effective in the business computer market. We *can* say that the design and manufacturing team has done its homework. ■

BOOK REVIEW

Artist and Computer

Edited by Ruth Leavitt
Harmony Books Division
Crown Publishers
New York NY, 1978
130 pages, paperback $4.95,
hardcover $10.00
Reviewed by G S Barrat

This collection of articles by thirty-five artists about themselves and their work describes and shows computer art that is almost always fascinating and sometimes stunning in effect.

Included are dozens of different ways of producing art with a computer, with a great many examples, some in color. Most of them require extensive computer resources, or complex optical and/or electronic systems. However, some of the techniques described could be used on a personal computer with a graphics capability, such as the Radio Shack TRS-80 Model I or the Commodore PET.

You could, for instance, copy the simple geometric patterns of Peter Struycken, or the kaleidoscopic colors of Joseph Scala, or the weaving patterns of Karen Huff, among others.

For those examples of a more complex nature, just enjoy the "Archimedean Spiral" by Ann Murray, the transformations of drawings and words by Leslie Mezei, the bird images by William Kolomyjec, the BLOCPIX of Ed Manning, and many others worthy of admiration.

In addition to being a fascinating book to dip into every now and then for a refreshing look at the beautiful and ingenious images that can be created with a computer, this book makes a fine present for artists, computer-oriented friends, or anyone who appreciates art. ■

Personal computers are turning up in the unlikeliest places—in this case, Southeast Asia.

an Apple in Hanoi

by Everett Hafner

At first the idea seemed wildly improbable. My assignment was to gather together the components of a scientifically useful microcomputer system and initiate myself into its deeper mysteries; then test it for a month, transport it to the Center for Scientific Research in Vietnam, assemble the system in a physics laboratory, and teach a selected group of scientists how to apply the equipment to their research. Easy enough, perhaps, if I was introducing a small system to a group of American scientists with consulting assistance as close as the nearest telephone, and spare parts no further than a neighborhood computer store.

The main obstacle to such a project was my own initial unfamiliarity with microprocessor technology. Although I am an experimental physicist with a fair working knowledge of digital electronics (I run an electronic music business), my prior computer experience had been gained exclusively at the terminals of timesharing systems whose processing units I had, in most cases, never seen. But I am reasonably quick in matters of this kind and I foresaw no difficulty in adding one more art to my bag of tricks.

Setting up a sophisticated system in Vietnam, however, was not going to be easy. As far as I knew, the only computers in the country were old IBM 360 systems abandoned by the American army six years ago and, most probably, lacking in parts. Since nothing similar to the American microprocessor revolution has swept the countries of the Socialist bloc (with whom Vietnam has good trade relations), I knew that there would be no trace of the products flooding the contemporary domestic computer market. The Vietnamese have no dollars to spare and, in any case, they cannot purchase computer equipment directly from us because of a United States embargo. As a result, Vietnamese technicians have no way of maintaining and repairing our computers without intensive training and liberal stocks of spare parts.

Another potential hazard was the likelihood of unfavorable environments. I was planning to travel in early March, just ahead of the rainy season. A summer spent in India had taught me how difficult it is to keep electronics running in the heat and wetness of Asian rains. There were certainly going to be problems with electrical power. The Vietnamese produce electricity at 220 V and a frequency of 50 Hz. It is simple to convert from one voltage to another with the help of a transformer, but the only way to obtain a new frequency is to generate it. Thus, if I wanted to be absolutely sure of what I was doing, I would need elaborate conversion equipment at either my home base in Massachusetts, or in Vietnam, depending upon the power requirements of my system. This was a dilemma: if I chose a system to match the foreign power supply, I had no simple way of testing it at home; if I chose to match the domestic supply, I would have to transform Vietnamese power on arrival. There was a considerable risk either way. It would be even riskier to assume that the line voltage in Hanoi might be sufficiently free of large fluctuations (not to mention frequent blackouts) for the system to operate without breakdown or error. Nothing could be more crucial to the success of the project. Unless the feeding hand is extremely steady, a computer has a strong tendency to bite it.

Another environmental problem grew to large proportions while I was preparing for the trip. In late 1978, Vietnamese forces entered Cambodia in support of a revolution against the incumbent regime. As a retaliatory move, Chinese troops subsequently crossed

About the Author:
Everett Hafner lives in Massachusetts, where he is Adjunct Professor in the School of Natural Science at Hampshire College, Visiting Professor of Physics at Williams College, and President of Electronic Music Studios of America.

the northern Vietnam border, destroying villages and industries as they came. The war would constitute a considerable distraction to my work. Perhaps I would find my scientific colleagues mobilized in support of the war effort. Many of my friends urged me to abandon the idea. Under the circumstances, they said, the last thing Vietnam needed was a microcomputer. I had to admit that the war was one more hazard on top of all the others I have described. In the late stages of my planning, a physicist for whom I have enormous respect, and whose work has taken him to other developing countries, spoke bluntly. "Everett," he said, "you're a smart guy with good intuition about probabilities. And you and I both know the probability that you can take a new computer to Hanoi, plug it in, and have it work. It's zero!"

It was indeed a time for retrospection. The idea was born when two Vietnamese scientists passed through Amherst on a tour of the United States. (They were the first citizens of Vietnam to obtain American visas since the fall of Saigon.) Their host for a day was Dr Arthur H Westing, Dean of the School of Natural Science at Hampshire College. Dr Westing is a botanist with a special interest in defoliant agents. He spent considerable time in Vietnam during the American war studying the ecological impact of our military programs. Dr Westing understood that the purpose of the Vietnamese visit was to generate support for their research institutes, and he suggested that I give them a tour of the extensive physics facilities in our academic community. It was a remarkable day.

I was pleasantly surprised to learn that physics had been active in Vietnam despite the hardships of recent decades, and that the people regard pure and applied science as an essential lever with which to work themselves out of their economic situation. There is a lively Institute of Physics in Hanoi, a wing of the Center for Scientific Research built with funds and technical assistance provided by the Soviet Union. Although much of its work is in pure physics, its principal purpose is to explore the country's rich mineral resources in the hope of developing a much needed export business. Some of the facilities at the Institute are modern and sophisticated. They include, for example, a technique for the detection of trace elements by neutron activation of mineral samples. At the end of the day, in conference with Arthur Westing, we raised the question of how to best enhance the technical capabilities of the Institute. The answer was immediate: a small system for computation and control of laboratory processes. I offered to take the job of assembling, delivering, and installing such a system. Dr Westing took on the task of raising the necessary funds.

In my first attempt to calculate a nominal budget for the project, I arrived at $15,000. This would cover my round trip air fare to Hanoi, a small living allowance en route (the Vietnamese were to be my hosts in their country), a few hundred dollars for excess baggage charges, a

fund for books and manuals, a supply of spare electronic parts, and the system itself. The first computer to come under consideration was the IBM Model 5110, a small and highly sophisticated system equipped with two high-level languages, BASIC and APL, selected by the flip of a switch. It was the APL facility that caught my attention. I have worked happily and productively with APL for ten years, and have developed an increasing respect for its power and its attractiveness to scientists. Its inventor, Kenneth Iverson, has been a friend over the years, and is always ready with stimulating suggestions for new applications of his language. We talked about the requirements for the Hanoi system. It became clear that the 5110 would do the job, and that the idea of introducing APL to a new group of physicists was extremely appealing. But it was gradually becoming apparent that our fund-raising effort was going to fall far short of my initial budget, forcing us to abandon hope of acquiring the 5110 or any other system in the $10,000 price range.

At this point I began to look seriously, for the first time, at the growing family of microcomputer systems. Having never been educated to appreciate the versatility of these machines, I shared what must be a common feeling among scientists brought up on large computers: the little systems were fine for hobbyists and small businessmen, but were far from powerful enough to meet the needs of a research laboratory. I was wrong. A fascinating conversation with Carl Helmers, Editor of BYTE magazine, started me on a delightful tour through the new world of microprocessor-based computers. I rapidly discovered that they are hard at work in laboratories close to home. Dr Joel Gordon, Professor of Physics at Amherst College, uses a Sol computer to control experiments in cryogenics; Dr Ken Williamson, chemist at Mount Holyoke College, has a Sol operating his lab. Dr Al Woodhull, biologist at Hampshire College, builds microprocessor systems for a variety of imaginative experiments, one of which tested the hypothesis that an eclipse of the sun enhances long-range propagation of radio signals. I was soon traveling from one computer store to the next, picking up a liberal education free of charge. My last stop, with only a month left before flying to Vietnam, was at the Retail Computer Center in Ludlow MA. By this time it was clear that our funds would level out at $6000, of which one-third would have to be reserved for travel expenses. When I sat down with Tom Yvon, owner of the store in Ludlow, the prospect of building a worthy system seemed terribly remote.

Tom helped me work out a plan (after whittling away some technological luxuries) that would provide full computer capability within my budget. I chose the Apple II for my central microcomputer, equipped with 48 K bytes of programmable memory and the Applesoft version of Extended BASIC firmware. In addition, I acquired two Apple Disk II floppy drives with ten spare diskettes, a Sony TC110B cassette recorder with ten cassettes, a Leedex Model 100 video monitor, a Centronics Model P1 Microprinter with ten spools of aluminized paper, a spare disk control board, and two unwired Apple input/output (I/O) boards. The power supply in my Apple was designed for 120 V at 60 Hz. A phone call to the Apple Company led me to believe that the supply would function at 50 Hz if I delivered an appropriate voltage with good waveform. Stability of the line voltage is important for these machines, but the actual voltage can lie anywhere between 107 and 132. All I needed to handle the problem, presumably, was a

high-quality step-down transformer. I chose the Triad Model N-7M unit which is designed to drop 500 W of power from 230 to 115 V. It had sufficient capacity to drive my peripherals as well.

I spent the entire month of February in my music studio, sequestered with the Apple II, working day and night on the enjoyable task of exploring the system, learning to program, and developing useful disk files. My only teachers were the manuals supplied with the system and the computer itself, which turned out to be its own best teaching aid.

I had previously found possible with typewriter terminals tied to large computers over which there is no direct control. My respect for microcomputers was growing by leaps and bounds.

The later stages of my education were devoted to mastering the machine language of the 6502 processor at the heart of the computer. [See "Understanding Personal Computer Software" by Louis Frenzel in the Winter 1979 issue of onComputing...ed.] Here again the Apple Company has been kind to its customers. The system contains a powerful monitor program that gives

Apple II system in operation, Institute of Physics, Hanoi

I had learned APL in the same way, using the responses of the system to educate myself to the rules and capabilities of the language. The two levels of BASIC in the machine constitute a powerful tool for handling problems in physics and mathematics. The graphic capabilities of the system are especially interesting. One of my first programs produces a vivid simulation of the tracks produced by a radioactive source in a cloud chamber; another is a "movie" of molecular motion, exhibiting a "random walk" of a particle in a two-dimensional gas. Another program shows the build-up of a complex waveform as successive terms of its Fourier components are added together. Other programs develop histograms of hypothetical experiments whose data follow Gaussian distributions, and exhibit elliptical orbits of planets around a star. All of these programs were easy to write and surprisingly concise. For fun, I invented a pinball game in which a ball scatters from pins as it falls down the screen under the influence of a "tilt" controlled by the player.

I also found a way of composing random music by displaying notes on a staff while sounding appropriate tones. The Apple II contains a small loudspeaker whose pitch is easy to program. Following this warm-up, I wrote a long program that calculates the best curve to fit to an arbitrary set of data points and then plots the result. Most fascinating of all, I wrote a program for graphing thousands of prime numbers on a rectangular spiral, exhibiting the mysterious 45-degree "rays" that appear in such graphs. By this time, I realized that I had passed far beyond the kind of mathematical exploration

immediate access to machine codes, and an efficient disassembler that enables the programmer to study machine programs that are already stored. With this facility, and the Apple II Reference Manual as an additional guide, I was able to enter regions of computer science that were entirely new to me. By the end of the month, I felt comfortable with such things as hexadecimal arithmetic, memory maps, pointers, stacks, op codes, low and high bytes, ASCII, data buses, and various modes of addressing. Whether or not I would be able to convey my information to colleagues who spoke little or no English was open to question. An even larger question still loomed before us. Would the system, performing so smoothly in my studio, run when I assembled it under less benign conditions halfway around the world?

At the beginning of March, just before I was to leave, the border war in Vietnam reached its height. I sent a cablegram to Hanoi suggesting a delay; their reply endorsed the suggestion. However, a week later, with the Chinese in retreat, we received another cable renewing the invitation. Along with the system, I packed a wealth of computer literature, and a supply of journals and chemicals that Ed Cooperman, a physicist in California, had sent to me for delivery to his colleagues in Hanoi. (Mr Cooperman had recently returned from the Institute where he spent a month installing a large gamma-ray-crystal detector in their neutron activation analyzer. He was one of many friends and colleagues who gave generous support to our project.) On the morning of March 15, I boarded the plane for the first part of my

"Eight reasons why Peachtree should be your source for micro business software."

1 We invented the wheel.

Founded in 1975, our team authored the first business software system for microcomputers. We invented the wheel so you don't need to re-invent it!

2 Over 1000 Installations

In our case, experience breeds success, and we have proven success with an installation base all over North America. The list of hardware vendors who either use or endorse our products reads like a Who's Who in microcomputers!

3 Feature-packed Systems.

We've built in more improvements and features than ever before. For example, our accounting system is integrated, or each module may be run separately in a standalone fashion. All systems have the features that experience has shown us to best fit the small business user.

4 We've included everything!

Our latest releases include such features as automatic file backup, recovery routines for hardware failures, and complete sample data bases for orientation and training. And with Peachtree SoftwareTM you receive the source code to make those necessary modifications!

TM

PEACHTREE
SOFTWARE

5 Our documentation is unsurpassed.

That's a pretty strong statement but time and time again when our Users Manuals have been compared with those of our competitors, ours have rated the best for accuracy, completeness and ease of use.

6 Software warranty.

Any programming defect reported in writing to Peachtree SoftwareTM will be corrected without charge for a period of one year after date of license. Unauthorized modification of software will void warranty.

7 We've immediate delivery!

And that means now! We have General Ledger, Accounts Receivable, Accounts Payable, Payroll, Inventory, Word Processing, Mailing Address and others. We're continually developing new products to increase our selection and, incidentally, if you have a marketable package, we're interested in talking to you about our distributing it for you.

8 You deserve the best.

Who can argue with that? We have the product, the track record, and the complete support program that will work for you. That's reason enough to deal with the leader!

For more information contact:

Marketing Department
Retail Sciences, Inc.
3 Corporate Square, Suite 700 • Atlanta, Georgia 30329 • (404) 325-8533

journey. Thirty-two hours later I landed in Bangkok, where I spent two days picking up a visa and seeing the sights while waiting for a flight to Vietnam. I reached Hanoi on the evening of March 19, met a small group of my new colleagues, and got acquainted over a ceremonial dinner. The next morning I delivered my computer and accessories to the Institute and met the rest of the physicists who were to work with me. I later joined a group of science administrators for *another* ceremonial feast of native food and drink. Finally, that afternoon, in a special lab of the Institute, my associates helped me unpack the system, assemble it, and verify the quality of the power source.

Then came the moment of truth. I turned on the video; the electron beam was scanning. I put power to the Apple; the monitor system was active. I quickly ran through a readout of the read-only memory; it was all in place. I then called up Integer BASIC; it accepted and executed the programs. I switched to Applesoft BASIC; ditto. After commanding the printer, it wrote faithfully. The disk operating system was activated, bringing up the disk loaded with my homemade program files. My colleagues were watching every step of the process with excitement, asking innumerable questions (mostly in Vietnamese), eager to learn how to do all of this and more. We were ready to begin our lessons. I had been in Hanoi for less than a day.

It was an instant of success worth dwelling upon. The idea that had been so quixotic a few weeks ago had suddenly become a reality. Working on a tiny budget, from an initial state of total ignorance, we had managed to bring up the first useful microcomputer in a scientifically active country of fifty million. (There is, in fact, a well-staffed but minimally equipped Institute of Computer Science and Applied Mathematics in Hanoi. At the time of my visit they had only reached the point of assembling a primitive Intel 8080 microprocessor for their own studies.)

In my haste to pack the system, I had forgotten to remove the Applesoft printed circuit card from its slot inside the computer. During the trip, with its episodes of rough-and-tumble baggage handling, I had imagined the board working loose and rattling around dangerously. When we unpacked, however, it was still tightly in place and ready to run. I must also mention the care with which the Institute had prepared for my arrival. A lab had been set aside for the computer, with air conditioning provided by a large Mitsubishi unit, as well as

Members of computer research team, Institute of Physics, Hanoi

voltage regulation equipment. In addition, the people working with me were well trained in physics and electronics. Each had spent a minimum of one or two years in excellent foreign universities, and in some cases had acquired a doctor's degree while in residence. In addition, they were enthusiastic, hardworking, and almost painfully hospitable.

The next few days were filled with essential tasks: class sessions and individual instruction, lectures to larger groups within the several institutes of the Center, and visits to labs elsewhere in Hanoi. My aim was to advertise the availability of the system for general scientific use, together with the services of my group as consultants. This implied, of course, that they would have learned everything they needed to know before I left the country. I had only a few weeks in which to bring them to that stage, and I was pleased to find that they were quick masters of everything I chose to discuss. Although I speak no Vietnamese, and few of them spoke English, the language barrier was surprisingly easy to overcome. During class sessions I had the help of an English-speaking member of the group. My larger lectures were translated by a government scientist educated at the University of Wisconsin. But more important than any human interpreter was the computer itself. Much of its vocabulary — graphs, functions, decimal numbers — is universal. Although BASIC has its roots in English, its short list of functions and commands is quickly absorbed by anybody. Thus I was able to teach by the same method that had helped me learn: write anything you wish into the system and study its response. The important skill is in knowing how to interpret a computer response, and selecting your next input. However, the ability to develop and use such a skill is independent of language.

Then came the moment of truth. I turned on the video; the electron beam was scanning. I put power to the Apple; the monitor system was active.

Just as we were making these forward strides, our luck ran out. On the afternoon of our fourth day, the young woman who was to manage the lab reported that the computer was responding strangely, or not at all, to system commands. I looked over her shoulder, and yes, there was trouble developing rapidly in the I/O sections: information was being garbled somewhere between the keyboard and the monitor, taking us out of communication with the processor. As we watched, the problem grew steadily worse until the system collapsed altogether, with no signs of life. It was the dreaded event, bound to occur sooner or later with any system, but striking us at a point where we had only started our work. In planning the project, I had decided that if such a thing happened I had no alternative but to pack up the system and bring it home. The only spare board in my hands was a disk controller, and we soon ascertained that this was not the cause of the failure. I had none of the integrated circuits that form the vital organs of Apple II, although most of them are fairly inexpensive. The aggregate cost of a full set (there are well over a hundred in the system) was far more than our budget could handle. Above all, my lack of training in computer electronics made the task of locating the problem difficult, if not impossible.

We let the system lie dormant for a few days while we debated various courses of action. During this period, I attempted to continue our class instruction on the blackboard, only to find that interactive dialog with the computer itself had indeed been a crucial element in the job I was trying to do. Without it I was reduced to the status of a foreigner struggling with the defects of word-by-word translation. In desperation I proposed that we go through the system with an oscilloscope, learning what we could along the way. I urged my group to study the block diagrams in the Apple II Reference Manual as an architectural outline. (The manual describes, in a general way, how information flows through the system. However, it gives no testing schedule and is therefore not in any sense a service manual.)

The following day things began to look better. The Apple power supply, the first component to come under suspicion, had passed careful tests of its voltages and its stability. With the help of an ancient Tektronix oscilloscope (another gift from Ed Cooperman), two engineers in the group traced the system failure to line 18 of the address bus, which was shorted to ground. An integrated circuit had failed. Fortunately it was a fairly common one. An hour later someone appeared with a replacement, installed it, and stood back to give me the privilege of firing up the system. I threw the switch. Marvelous! The entire computer came back to life. Our shouts drew people from surrounding labs, a bottle of coffee liqueur appeared from somewhere, and we offered toasts.

I later asked for an account of the episode for my records. Neither of my two engineers spoke or wrote in English; they produced an account in Vietnamese and handed it to the young physicist who had been interpreting for me. The following is his translation:

> "March 22 1979. Afternoon. Viet Dzung and Thi Minh, trying BASIC language, see that the relation between the video and the keyboards is no good. March 23 1979. Morning. Professor E Hafner works in lab. The computer works well only for about a half hour, then something is wrong again. At first we think that the 50 Hz frequency of the power and the temperature in the room influence the machine's operation. March 28 1979. Morning. After testing

RAMs with R/W commands, Mr Cong and Mr Sat find all RAMs good and the power supply can't have failed. Viet Dzung, measuring and testing, finds the IC 74LSOO failed...Afternoon. He changes this IC by 74OOSN. It is OK! The Apple II works completely well. Everybody shouted with joy. Opened wine. How happy! How happy!"

I was deeply impressed by this performance, which far outshone anything I could have done alone in such a short time. It is true that my engineers had had formal training as students of computer science. Viet Dzung, for example, had attended the Polytechnic Institute in Hanoi, and Manh Sat had attended the Center for Nuclear Physics in Dubna (Soviet Union). Nevertheless, their skill in diagnosing the shorted circuit in our system was astounding. With no more than a layout and flowchart of a strange new system, and with antiquated test equipment, they had burrowed into the root of the problem in a few hours.

When I returned to Massachusetts after my five-week visit, I found time to reflect on the experience. First, something that has nothing directly to do with Vietnam: I had become vividly aware of the sophistication that exists in modern microcomputer systems. Without having been forced by financial constraint to look carefully at such systems, I would have missed a rich experience. Indeed, if money had been more plentiful, we might easily have chosen a larger, more powerful system, but one that would have been less feasible to maintain at an isolated and poorly equipped site. I knew when I left Hanoi, that the system was in good hands, and that there was every reason to hope for its long life and continuing utility. More generally, I left with renewed faith in the ancient idea that scientific cooperation is one of the most effective ways of forging links between cultures, even in the presence of sharp political differences. The step I was able to take—microscopic in the context of our own computer development, but very large indeed to the Vietnamese—will presumably be followed by many others when relations with socialist Southeast Asia have stabilized. Until that happens, the only way to work with their scientists is through the kind of private action that I was able to take.

I miss my many friends in Hanoi, but there is one friend that I need not do without. A few days after my return, I went to see Tom Yvon in Ludlow once again, and purchased an Apple of my own. It continues to be not only a good friend, but one of the best teachers I have ever had. ■

BOOK REVIEW

An Introduction to Personal and Business Computing

by Rodnay Zaks
Sybex
Berkeley CA, 1978
paperback $6.95
Reviewed by Phil Hughes

An Introduction to Personal and Business Computing by Rodnay Zaks is an excellent book for anyone new to computing and electronics who is considering a microcomputer for either business or personal use. The first chapter describes a day in the electronic city of the future to illustrate the possible applications of microcomputers. The next two chapters describe the running of a program on what could be considered an average microcomputer of today and then define computer terms and jargon.

For those who want more information about hardware, chapter 4 covers system architecture. This material can be omitted without affecting the readability of the remainder of the book.

The next two chapters deal with programming. Chapter 5 presents some basic concepts, including languages, flowcharting, the ASCII code, logical operations, and program development. Chapter 6 deals specifically with the BASIC and APL languages. The general structure of BASIC is presented using short sample programs. Some tradeoffs and limitations are then presented, and finally APL is briefly explored as an alternative to BASIC.

Chapter 7 explains what functions a microcomputer can perform for a small business and shows the relationship between different programs and files. The chapter ends with a checklist of business software facilities.

The next three chapters discuss system selection. Both the microcomputer itself and peripheral equipment are discussed. These chapters discuss existing hardware as of late 1978.

The last four chapters discuss the economics of a system, possible failures, where to get help, and a short discussion of what the future holds.

For those looking for specifics, there are six appendices dealing with computer logic, bits and bytes, data communication, files and records, and a list of small business system manufacturers and microprocessor manufacturers. ■

8080 Assembly Language for the Totally Lost

by W D Maurer

So you've fooled around with BASIC for a while on your new 8080-based system, and are getting slightly bored with that. You feel it's time to try assembly language. After all, you say to yourself, this can't be hard; it's only another computer language; I already *know* a computer language, namely BASIC; they're all much the same, with GO TOs and IFs and plus and minus and so on. Right?

So you start reading a beginning assembly-language book and immediately you discover that yes, all computer languages are most definitely alike in some ways. There is ADD, which corresponds to +; there is JMP, or "jump," which corresponds to GO TO; there is MOV, or "move," which looks like an admittedly rather simple assignment of the form LET A=B or just A=B, depending on which version of BASIC you have. There are also several mysterious looking things called DAD and POP, about which you are sure only that neither of them has anything to do with your father; but, after all, not everything in BASIC was obvious the first time you read it, either.

If you're like many of us, you read about the 8080, and then you read about it again, and still again, and put the book down with that unmistakable "something's missing" feeling. Or, more explicitly, it's that "something's missing but I don't know what it is" feeling. More specifically, some of the questions you might be asking yourself are the following:

(1) I can clearly ADD A, or B, or C, or D, E, H, L, or M. But suppose I want to ADD something else? What do I do then?

(2) I can clearly subtract by using SUB, but what do I do if I want to multiply or divide?

(3) How do I set Q7 (let us say) equal to some expression? There doesn't seem to be anything on that subject.

(4) What is there in assembly language that corresponds to FOR I = 1 TO 10 and NEXT I?

(5) How can I have an instruction that just says *compare*? The word "compare" means so many things; you could be comparing to see if two quantities are equal, unequal, or whether one is greater than the other, less than the other, and so on.

(6) After reading the description of PUSH, I *think* I understand what it does; but why would you ever use such a thing?

W D Maurer is professor of computer science at George Washington University in Washington DC, and author of several books, including Programmer's Introduction to SNOBOL. *He is also a frequent contributor to* BYTE *magazine.*

This article is written with the assumption that the reader knows an algebraic language, namely BASIC. I'll confine myself to explaining one assembly language, namely that of the 8080, partly because I feel it is a bit more difficult to understand than that of the 6800 or the 6502.

To begin with, you need to understand about *registers*. These can be thought of, from the point of view of the BASIC programmer, as simply special variables with special properties. The process of setting a variable to a new value, when you are using assembly language, normally (that is, if you don't cut any corners) consists of three parts:

(1) First you take your variables and move them into the registers. This process is known as *loading.*

(2) Once your variables are in the registers, you can add and subtract them and do various other things to them, always in the registers, until you have the result you want.

(3) Once you have the result you want, it will be in a register, and you have to move that to a place which denotes the value of a variable. This process is known as *storing.*

Let us consider the setting of J equal to K + 3. In assembly language, we would do three things. We would *load* K, using the instruction LDA K

(load the A register with K). We would then add 3, using the instruction ADI 3 (not ADD 3; this will be explained later). This adds 3 to K, which is already in the A register, and leaves the result, namely K + 3, in the A register. Finally, we would *store* this result in J, using the instruction STA J (store the A register in J). Thus the three instructions:

```
LDA   K
ADI   3
STA   J
```

give the effect of J = K + 3 in BASIC.

The 8080 computer has nine basic registers. For the moment, let us consider the first five, which are called A, B, C, D, and E. Of these five, the A register is the *only* one which can be added to, subtracted from, or even (without using registers other than these five) loaded or stored. Let us see what this means when we try to set J = K + N, rather than J = K + 3. The addition of K and N must take place in the registers; let us assume that the value of K is in the A register, and the value of N is in the B register. Then we can do an instruction, ADD B, which will add the B register to the A register and leave the result (in this case K + N) in the A register. Then we can do STA J, just as we did before, to set J equal to this number.

But how do we get K and N into the A and B registers in the first place? The instruction LDA K loads the A register with K, but we don't have an instruction to load the B register. We do, however, have an instruction to *move* the A register into the B register. This suggests the following sequence:

(1) Load the A register with N (LDA N).
(2) Move the A register to the B register (MOV B,A).
(3) Load the A register with K (LDA K).

(Notice that we have to do N first, because if we did K first, then, when we loaded N into the A register, K would not be in the A register anymore.) Thus the five instructions:

```
LDA   N
MOV   B,A
LDA   K
ADD   B
STA   J
```

give the effect of J = K + N in BASIC.

Isn't there a better way? Yes, there is; but it involves some of the other registers.

Let us go back to the difference between ADD and ADI. When we have a *constant* to add, such as 3 (the constant must be between 0 and 255, inclusive), we use ADI. When we want to add what is in a *register*, we use ADD. The register is always added to the A register, *and the result always goes back in the A register*. The register we add can, itself, be A; that is, we can ADD A, which adds A to A and puts the result in A — that is, it doubles the quantity in the A register.

But how do we add a quantity like N, other than putting it in a register *like the B register* first? There is a general, all-purpose instruction for this, called ADD M. You can use ADD M to add *any* variable whatsoever to the A register, and leave the result in the A register. However, before you do this, you have to tell the system what variable you want to add. This is done by using two more registers, the H and L registers, and (usually, although not always) an instruction called LXI.

The instruction LXI H,N sets up the H and L registers to indicate that N (rather than any other variable) is the variable to which ADD M, and other instructions involving M, refer. Once this has been done, we can ADD M and the result will be the same as adding N. Thus we can do:

```
LXI   H,N
LDA   K
ADD   M
STA   J
```

to give the effect of J = K + N, using only four instructions instead of five.

To understand the LXI a bit better, it is necessary to consider, for a moment, the sizes of our registers. Each register (A, B, C, D, E, H, or L) can hold any number from 0 to 255 inclusive. This is because each register holds eight bits, or binary digits, and $255 = 2^8 - 1$. (In the same way, if the digits were decimal instead of binary, the largest possible number in eight digits would be 99,999,999, which is $10^8 - 1$.) When you put two registers together, like H and L, and each one contains eight bits, the combined, or *double*, register will hold 16 bits, and the largest number it can contain is 65,535, which is $2^{16} - 1$.

The number 65,536, or 2^{16}, is often abbreviated as 64 K, since it is 64 times 1024. This is the largest number of individual cells that can be in the memory of an 8080. These cells contain variables (such as J, K, or N), elements of tables, codes for characters (such as letters and digits), and codes for instructions in programs. Each cell has an *address*, which is a number from 0 to 65,535. The address is like a cell number (as in "cell 0, cell 1, cell 2," and so on). We cannot put an address in the A register, in general, because the A register will hold only a number from 0 to 255. But we can put an address in the HL register pair (that is, in the double register formed out of the H and L registers); and, when we do so, ADD M will add

whatever variable is in the cell that has that address.

The instruction LXI H,N puts the *address* of N in the HL register pair. It is important not to confuse the address of N with the value of N. Every variable has a value between 0 and 255 (for single-byte variables). Every variable, however, is also kept in some cell in the memory, and that cell has an address, which we call the address of the variable. This never changes during the running of the program, whereas the value of the variable, of course, can change many times (if it is kept in programmable memory).

The notation M does not apply only to adding. There is an instruction MOV A,M which loads the A register with whatever is in memory "at the location M" (that is, in the cell whose address is currently in the HL register). This kind of loading applies to *any* register we have discussed; that is, there is MOV B,M, MOV C,M, and so forth. There is also MOV M,A which *stores* the A register into the cell in memory that is at the location M; and there is likewise MOV M,B, MOV M,C, and so forth.

Once the HL register pair has been loaded, it does not need to be reloaded with the same quantity. Consider the instructions:

```
LXI   H,J
LDA   K
ADD   M
MOV   M,A
```

The first of these puts the address of J in the HL register pair. The next one puts K in the A register. The next one adds J, because the HL register contains the address of J, and puts the result, namely K+J, back in the A register. But the HL register pair still contains the address of J, and so the last instruction moves A — that is, K+J — into memory at this location; that is, it gives J this new value. The result is the same as J = K+J (or J = J+K) in BASIC. (Of course, we could also have written just STA J instead of MOV M,A.)

Now that we understand this much about assignments, let's try IF statements. Suppose we want to do the equivalent of IF X+Y=Z THEN 100. First we have to add X and Y and compare the result with Z. This much can be done by:

```
LDA   X    ; PUT X IN REGISTER A
LXI   H,Y  ; PUT ADDRESS OF Y IN HL
ADD   M    ; ADD Y TO X (X+Y TO A)
LXI   H,Z  ; PUT ADDRESS OF Z IN HL
CMP   M    ; COMPARE X+Y WITH Z
```

One nice property of assembly language is that we can put a comment or remark on *every* line. (The 8080 assembler used by this author requires that every such comment be preceded by a semicolon.) This allows sequences of assembly-language instructions to be more or less self-explanatory, as above.

But what does the CMP (compare) instruction do? It performs *four* different kinds of comparisons at once. The result of each of these comparisons is set by CMP in a single bit (binary digit) of the status register, a register we have not looked at so far. There are then eight different kinds of conditional jump instruction, two for each kind of comparison. One of the two (for each kind of comparison) does a jump (equivalent to "going to" a label) only if the result of the comparison is "true." The other one of the two does a jump only if the result of the comparison is "false." The four kinds of comparison are as follows:

(1) *Zero.* The zero flag in the status register is set to 1 on an *equal* comparison, and to 0 on an *unequal* comparison. The instruction JZ (jump on zero) jumps if the zero flag is 1; the instruction JNZ (jump on nonzero) jumps if the zero flag is 0.

(2) *Carry.* The carry flag in the status register is set to 1 on a *greater than or equal* comparison (of unsigned numbers from 0 thru 255), and to 0 on a *less than* comparison. The instruction JC (jump on carry) jumps if the carry flag is 1; the instruction JNC (jump on no carry) jumps if the carry flag is 0.

(3) *Sign.* We have so far not mentioned the fact that it is possible to treat a variable as a signed number between −128 and 127, inclusive, rather than an unsigned number between 0 and 255. For such numbers, the sign flag in the status register is set to 0 on a *greater than or equal* comparison, and to 1 on a *less than* comparison. The instruction JP (jump on plus) jumps if the sign flag is 0; the instruction JM (jump on minus) jumps if the sign flag is 1.

(4) *Parity*, which in this introduction can be skipped. (There is also a fifth comparison, auxiliary carry, which has no associated jumps, and which will also be skipped now.)

The result of all this is that you almost always follow a CMP (compare) with one of the conditional jumps (JZ, JNZ, JC, JNC, JP, or JM), and the result is the same as if you had said "if the condition holds, then go to some location." The location to which we go is given along with the conditional jump; thus if we write:

```
CMP   M
JZ    ALPHA
```

this means "If the A register equals the quantity in memory at the location M" (as this is defined

above) "then go to ALPHA." In BASIC, by the way, we always go to a line number, but in assembly language we go to a point with a *label* like ALPHA appearing in the lefthand column (as a line number would), followed by a colon. Labels and variable names in assembly language both follow the same rules: they must start with a letter; they contain only letters and digits; and they cannot be more than six characters long. (Any string of characters satisfying these rules is called an *identifier*. In some assembly languages, these rules are tightened or relaxed a bit, but the first of them always holds.)

Since we cannot go to 100, let us go to L100 instead, where L100 (followed by a colon) appears in the first five positions in line number 100. Then our IF statement is translated as:

```
LDA    X       ; PUT X IN REGISTER A
LXI    H,Y     ; PUT ADDRESS OF Y IN HL
ADD    M       ; ADD Y TO X (X+Y TO A)
LXI    H,Z     ; PUT ADDRESS OF Z IN HL
CMP    M       ; COMPARE X+Y WITH Z
JZ     L100    ; IF EQUAL, GO TO L100
```

FOR statements in BASIC reduce to IF statements in the following way. Suppose we have a FOR loop as in listing 1, specifying the repetition of statements α through β. Then the loop of listing 2, which does not use any FOR statements, does the same thing. (We can use the same idea in more general FOR loops, where the statement FOR I = J TO K STEP L is used, if we simply replace I = I + 1 in listing 2 by I = I + L.) Thus if we know how to translate IF statements into assembly language, we automatically know how to translate FOR statements.

```
        FOR I = J TO K
α               (Statements α through β)
        .
        .
        .
        .
        NEXT I
```

Listing 1: A FOR loop written in BASIC.

```
        I = J
α               (Statements α through β)
        .
        .
        .
        .
        I = I + 1
        IF I < = K THEN α
```

Listing 2: This loop, also written in BASIC, does the same thing as the FOR loop in listing 1.

No treatment of FOR statements would be complete, however, without discussing subscripted variables. Suppose we have an array A with elements A(1), A(2), A(3), and so on. We then have the question of how to load, store, and do various other things to A(3), for example, and also to A(I) for some program variable I (which is a different problem entirely). We will now give procedures for doing this.

Let us make the assumption that all of the elements of our array are stored sequentially. What this means is as follows. Suppose that A(1) is stored in the cell that has address α. We would like to store A(2) in the cell that has address $\alpha+1$, and then A(3) in the cell with address $\alpha+2$, and so on. This means that A(I), for any value of I, will be stored in the cell whose address is $\alpha+I-1$ (or equivalently, as it is usually written, $\alpha-1+I$). We should mention that this assumption is almost universal, and that the cases in which it does not hold involve advanced programming techniques anyway.

Loading A(3) (or A(7), or whatever) is now a matter of calculating the correct address. In assembly language, if we wanted to load A(3) with an LDA instruction, we would write LDA A+2 (or, for storing, STA A+2). This is *highly* confusing to many people the first time they see it. For one thing, it looks like what we are loading is the *value* of A plus 2, or 2 more than the value of A. In fact, we cannot do this with a single instruction; we would have to load A first (if it were not an array) and then add 2 (using ADI 2) with the next instruction. What LDA A+2 does is to load from the cell whose address is $\alpha+2$, as we have defined it above — that is, from the cell whose address is found by taking the address of the cell A and then adding 2. We can also put $\alpha+2$ into the HL register by doing LXI H,A+2 — and when we do this, M refers to A(3), so that we can load A(3), store it, add it, subtract it, and so on, by using the designation M as discussed earlier.

Working with A(I) is quite different. In the first place, the *only* way we can work with A(I) (or A(u) for any variable or expression u), without using advanced techniques, is to get its address into the HL register. Suppose we have calculated the value of u and put it (for a reason to be described in a minute) in the E register. (If it is already in the A register, this is easy enough; we just do MOV E,A.) By the formula given earlier, we now have to take this quantity and add $\alpha-1$, if we do so, we will get the address of the cell containing A(u). The trouble is that $\alpha-1$, is a 16-bit quantity; it will not fit into a single register, and we cannot add it to anything using an ADD instruction, which acts on single registers only.

This is where we use DAD, which stands for "double add." If we have two 16-bit quantities, one in the HL register pair and one in the DE register pair, then DAD D will add them, as 16-bit quantities, and put the result in the HL register pair. This is exactly where we want the result in this case, of course. There are only two more things we have to do before the DAD. The first is to load the HL register pair with $\alpha-1$, which we can do with LXI H,A−1. The other is to set the D register to zero, which we can do with MVI D,0. The reason the D register must be zero is that then the DE register (that is, the D and E registers, taken as a double register) will contain a 16-bit quantity which is the same as the 8-bit quantity now in the E register.

Thus, for example, if J is in the B register, we could put A(J) in the C register by the following sequence:

```
MOV   E,B   ; PUT J IN THE E REGISTER
LXI   H,A-1 ; ADDRESS OF A MINUS 1 TO HL
MVI   D,0   ; CLEAR THE D REGISTER
DAD   D     ; ADDRESS OF A(J) TO HL
MOV   C,M   ; A(J) TO THE C REGISTER
```

Finally, a word about subroutines (and, while we are at it, multiplying and dividing). In BASIC, you say GOSUB *n* to go to a subroutine at line number *n*. In assembly language, the subroutine will have a name, like SUB, and you say CALL SUB. The subroutine then starts with the label SUB. At the end of the subroutine, where you would write RETURN in BASIC, you write simply RET in 8080 assembly language. (The 8080 also has conditional returns, which are like conditional jumps. Thus RZ, or "return on zero," is like JZ ALPHA, or "jump on zero to ALPHA," except that it returns instead of jumping.)

A general rule about subroutines is that, if you don't have a single instruction on your computer that does what you want, then use a subroutine instead. In particular, this applies to multiplication and division. There are very few microcomputers that have multiply and divide instructions (although almost all big computers have them), and the 8080 is certainly not one of them. If you need to do multiplication or division, you will have to get hold of a subroutine (there are several in the literature) and then call it when necessary.

Registers often cause problems when a subroutine is called. If you have something (call it *x*) in the HL register pair, and you call a subroutine, then, when the subroutine returns, *x* might not be in the HL register pair any more, because the subroutine might have needed the HL register pair for its own calculations. For this reason, it is good practice for a subroutine to start off by *saving* the registers it uses, and to end up by *restoring* these registers to the contents they held before. Saving is done with PUSH, and restoring is done with POP. Thus if you have a subroutine which uses the D, E, H, and L registers, you would write:

```
PUSH D  ; SAVE DE REGISTER
PUSH H  ; SAVE HL REGISTER
```

at the beginning of the subroutine, and:

```
POP H   ; RESTORE HL REGISTER
POP D   ; RESTORE DE REGISTER
```

at the end (remembering to pop in the *reverse* order of pushing).

There is much, much more to assembly language than this, of course. But remember, this article was for the *totally* lost. After reading the article, you should still be lost, but at least you should know how, with the help of further literature, to find your way from here. ■

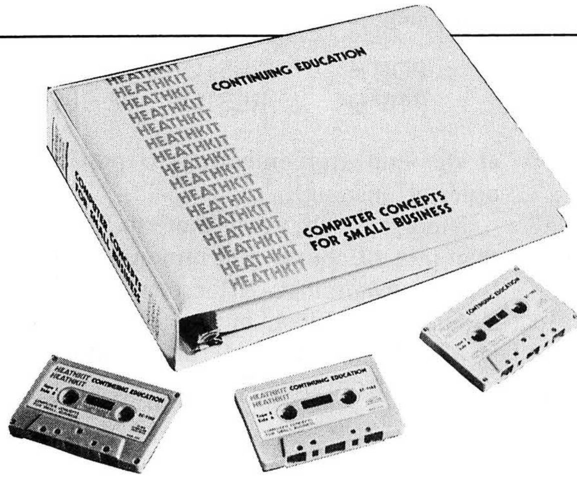
REFERENCE

Weller, WJ et al, *Practical Microcomputer Programming: The Intel 8080*, Northern Technology Books, Evanston IL, 1976.

New Products

Conducted by Charles Freiberg

These pages are designed to keep our readers in touch with the marketplace. The material which appears here is obtained from manufacturers and is not to be taken as an endorsement by onComputing. We invite manufacturers to submit material and we publish the information we feel will be of interest to our readers.



Program Helps Business People Evaluate the Utility of Computers

Computer Concepts for Small Business is aimed at the businessperson who knows little or nothing about computers and who wants to determine how a computer can reduce costs and increase efficiency. The program, including three audio cassettes and a 160-page workbook, describes the types of memory in a computer, compares the capabilities of different types of storage media and ports, and discusses the types of software and the tasks that they can be designed to perform in a typical small business. It also covers the personnel needed to run a computer, how to set up a data processing department, what to look for in a data processing manager, the advantages of permanent versus free-lance data processing employees, and describes the various advantages of time-sharing versus service bureaus versus ownership of the computer. The program costs $49.95. For more information, contact Heath Co, Dept 350-970, Benton Harbor MI 49022.

Circle 259 on inquiry card.

Computer Bridge Game

Tryom Inc, 23500 Mercantile Rd, Cleveland OH 44122, has developed the specifications for The Charles Goren Bridgemaster, a computer bridge game. The program features include the ability to play any number of hands from one to four. It can also function as a scorer when no computer hands are played, play on multiple skill levels for beginners and advanced players, enter a hand into the computer for use as a tool for examining specific situations, and be a cumulative game scorekeeper. The game uses encoded playing cards with two decks supplied. The programming for the Bridgemaster is designed for duplicate bridge. Bridgemaster does not take vulnerability into account when making decisions. This distinction is one of the highest orders of bidding refinements and is not in the program's scope.

Circle 262 on inquiry card.

Power Supply for Power Failures

Mayday provides instant power switching when your line voltage fails. It maintains the program in the computer and saves the program or data on a disk. Mayday is designed for the Radio Shack TRS-80, and can keep a video display, expansion interface, microprocessor, and four disk drives operating for about one half hour of power outage. Mayday will also handle most microcomputers. The unit maintains charge on the standby battery during normal use and can use a normal automotive battery for extended use of the complete system during power outages. The entire unit costs $240 and is available from Sun-Technology Inc, POB 210, New Durham NH 03855.

Circle 261 on inquiry card.

Text Processor for Apple II and Apple Plus Microcomputers

Charles Mann and Associates, 7594 San Remo Trl, Yucca Valley CA 92284, has released a text processor for the Apple II and Apple Plus systems called the Master Text Processor. The system includes a mailing list element and a programmable form letter writer. It operates on 32 K bytes of programmable memory and either one or two disk drives. The editing routine package includes change operations, insert operations, delete operations, text reformatting, and more. The built-in mailing list element allows for a five-line address field, provision for the new ZIP code fields, a code field usable in sorting, list printing and form letter writing, and a user field. The form letter element is designed to find letter recipients under user control, select the correct letter format for the recipient, and printer the letter with the proper form of address. The Master Text Processor is available for $139.

Circle 263 on inquiry card.

Common BASIC Problems on Book, Cassette, and Floppy Disk

This book was designed for people who use a variety of practical BASIC programs. It contains listings and documentation for seventy-six programs that can be used on many microcomputer systems. The documentation is complete so that only minimal programming knowledge is required to run the programs. Examples from the table of contents include: mortgage amortization table, linear interpolation, integration, linear programming, multiple linear regression, and tax depreciation schedule.

The programs also come on cassette and floppy disk for the TRS-80 and PET microcomputers. The book costs $12.50 and cassettes are $15.00, while the PET disk is $22.50. For more information, contact Osborne/McGraw-Hill, 630 Bancroft Way, Dept W19, Berkeley CA 94710.

Circle 265 on inquiry card.

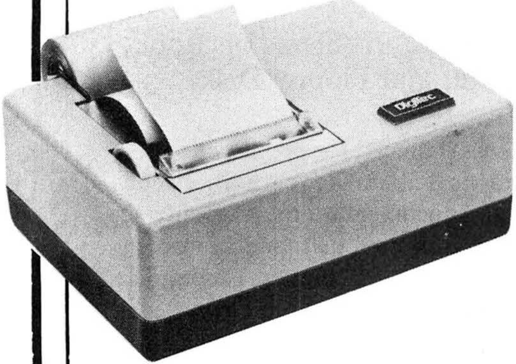
Automated Correspondence System for Businesses

The Cyberscribe II system simplifies and speeds the creation of letter correspondence. It can create correspondence quickly and accurately from a data base of preformatted letters or paragraphs or from an on-line correspondence data base in response to requests made by users at video display terminals and as users compose one-time letters at video display terminals. In all cases, a minimal input of operator supplied variables and keystrokes is required and clerical data searching time is eliminated. Other features of this system include terminal operator prompting, correspondence routing, multiple original letter preparation, and multiple letter copy preparation. It can accommodate any size forms and runs on any IBM-compatible computer system, Model 360, 370, 30xx, or 43xx. The software is written in Assembler and ANSI COBOL.

For more information on price and availability, contact Cybertek, 6133 Bristol Pky, Culver City CA 90230.

Circle 264 on inquiry card.

New Products



Alphanumeric Thermal Printers

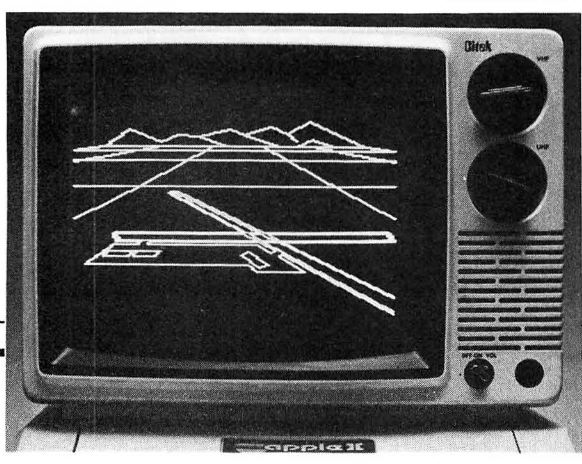
Priced at approximately $440, the 6450 and 6460 alphanumeric Thermal Printers produce easy-to-read letters, numbers, and symbols on thermal paper. They print sixty-four characters with 21 characters per line and approximately 6500 lines per paper roll. The Model 6450 provides a serial input with selectable RS-232C or 20 mA current loop with data rates of 110 and 300 bits per second. The Model 6460 is 8-bit parallel bus compatible with data rates up to 1000 characters per second. Both models respond to ASCII input format. Contact United Systems Corp, 918 Woodley Rd, Dayton OH 45403.

Circle 266 on inquiry card.

Apple II Animation Package

The A2-3D1 is a package of easy-to-use assembly-language programs for 3D and 2D animation on the Apple II computer. The program allows users to view two- or three-dimensional scenes created in the standard XYZ coordinate system, zoom between wide angle and telephoto fields of view, and select a location in space and a direction of view. One feature allows users to generate an output array of line start and end points, instead of plotting on the Apple screen. Other features include zero-page restore, which leaves all zero-page variables intact after subroutine exit, and page control for selective page erase, display, and draw for ping-ponging between screens for smooth animation. The load and go manual guides beginners through an orientation session with the A2-3D1 program. The technical manual is for advanced applications and describes the transformer algorithm in detail. The program requires 16 K bytes of programmable memory for 3D to 2D transformer, small scenes, and small control programs. Larger scenes, control programs, and the DEVELOP program require 24 K bytes of programmable memory. The program costs $45 on cassette and $55 on floppy disk. For more information, contact Sublogic, POB 5, Savoy IL 61874.

Circle 267 on inquiry card.



APF Electronics Introduces the Imagination Machine

APF Electronics Inc, 444 Madison Ave, New York NY 10022, has introduced the user programmable Imagination Machine. The Imagination Machine's educational tutoring network aids students in learning and developing potential skills. This interaction feedback network gauges the user's learning pace. Standard features include color, 10 K bytes of read-only memory and 9 K bytes of programmable memory, typewriter type keyboard, fifty-three keys with a shifter "BASIC Keywords" button, two game style controllers, 32 characters by 16 lines screen format, six-function built-in cassette deck, built-in microphone jack, cartridge connector, plus an Internal Operating System and BASIC Interpreter.

Circle 268 on inquiry card.

Protect Your Disks from Mail Damage with Floppy-Armour

Floppy-armour is tough, lightweight protective packaging for either 5-inch or 8-inch floppy disks. It is clean and white and designed to protect the disk from folding and creasing damage. First-class postage is $0.41 for the disk in the Floppy-armour package. The cost for the 5-inch disk version is $0.60 each and for the 8-inch version the cost is $0.90. The minimum order is one hundred. For further information, contact Square 1, 614 Eighteenth Ave, Menlo Park CA 94025.

Circle 269 on inquiry card.

New Products

Keyboard Enhances Operation of Commodore PET

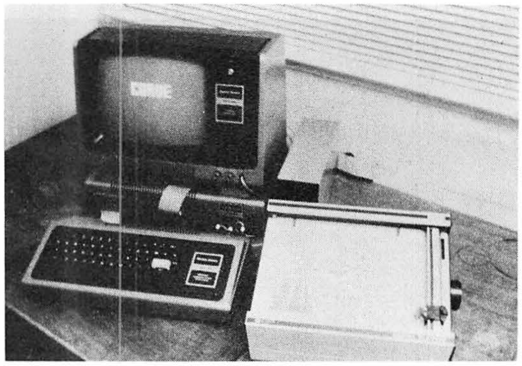
The Professional Encoded Remote Keyboard (PERK) for the Commodore PET computer is a plug-in, typewriter style, alphanumeric keyboard, designed to enhance the operation of the PET. The PET comes with a built-in calculator type keyboard, but the keys of the built-in keyboard are only half the size of standard typewriter keys. The PERK standard keyboard shares the PET internal keyboard interface, allowing the two keyboards to be used interchangeably. Both are active at all times, allowing the operator to use the PERK keyboard for normal data entry, and the PET keyboard for numerics or PET's extensive graphic capabilities.

Housed in a custom steel desktop enclosure, the PERK is connected to the PET by means of a plug-in interface card. Once installed, the PERK remote keyboard is immediately usable on all existing software. No changes or modifications are required, and both internal and external keyboards may be used simultaneously.

The PERK keyboard gives the user complete cursor control as well as the ability to edit text on the screen.

The PERK keyboard is $229.95, wired and tested. Full instructions, reference card, and 90 day warranty are included. For further information, contact George Risk Industries Inc, GRI Plaza, Kimball NE 69145.

Circle 271 on inquiry card.

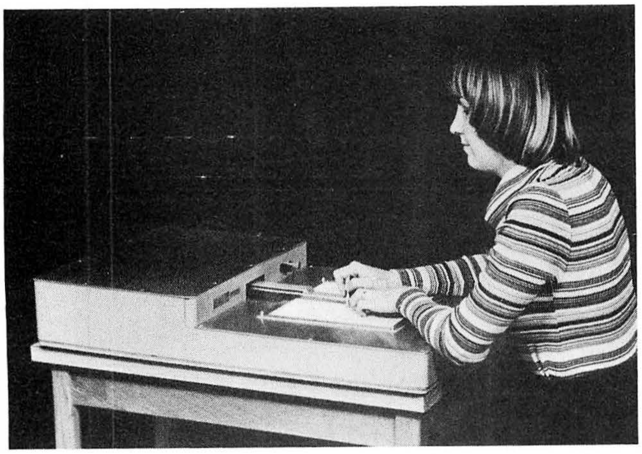


Hardcopy Graphics for the PET, TRS-80, and Apple II

West Coast Consultants, 1775 Lincoln Blvd, Tracy CA 95376, has released Curve, a program which provides hardcopy graphics capability for Houston Instruments' HIPLOT plotter. The program drives the low-cost plotter through an RS-232 interface. The program is written in BASIC and is capable of plotting Cartesian, polar, and parametric equations, as well as data points from the keyboard. It also features the ability to plot bar graphs with or without shading and it has built-in error messages that trap bad or unreasonable input quantities. A fully scaled alphanumeric character set is available to the user in two font styles. The prices range from $50 to $80.

Circle 272 on inquiry card.

New Products



Sensory Quill Enables the Blind to Write and Draw

A writing and drawing device for the blind or partially sighted professional has been introduced by Traylor Enterprises Inc, 830 NE Loop 410, Suite 505, San Antonio TX 78209. The Sensory Quill is electrically operated and allows an individual freedom to write or draw on paper in raised lines that can be felt. The instrument enables a blind person to write, draw, feel, and interpret everything from simple copy to complex statistical graphs and diagrams, maps, blueprints, and organizational charts. The machine also enables computer programmers to do flowcharting. It will produce metal masters for multiple copies of raised line materials.

The institutional model sells for $795 and the portable home model is priced at $425.

Circle 273 on inquiry card.

Anti-Glare Device for Video Screens

Sun-Flex Company is offering a black woven nylon mesh stretched on a flexible plastic frame designed to be sandwiched behind the video bezel, and to conform to the surface of the tube. This device blocks and absorbs ambient light with a honeycombing effect. The contrast is enhanced by the black matrix effect of the fabric background; the display characters are transmitted undistorted through the pores in the material. The filters are available in 120 sizes. Each size can be equipped with different optically graded fabrics to vary the intensity of the video display. The filters improve the image, lower maintenance, and reduce eyestrain and related stress. For more information, contact Sun-Flex Co Inc, 3020 Kerner Blvd, San Rafael CA 94901.

Circle 275 on inquiry card.

Compucolor II

Compucolor Corp, POB 569, Norcross GA 30071, has developed three models of the Compucolor II. Model 3 has 8 K bytes of programmable memory, the Model 4 has 16 K bytes, and the Model 5 provides 32 K bytes of memory. The Compucolor II uses an 8080A microprocessor and includes 16 K bytes of read-only memory. One RS-232C serial port is provided for a printer or modem. The Compucolor II features a separate keyboard and video display. The video terminal commands include erase line and page; two character sizes; fifteen plot modes; local, full, and half duplex modes; full cursor control; and other functions. The system uses Disk BASIC 8001 with an interpreter in read-only memory. Twenty-nine statement types, three command types, nineteen mathematical functions, and nine string functions are included.

One 5-inch floppy-disk drive is built into the video terminal. The capacity for each side of a 5-inch disk is 51.2 K bytes of memory.

The video display features eight colors with 32 lines and 64 characters per line. The usable screen area is 23 cm (9 inches) wide by 17 cm (6.75 inches) high. Compucolor has developed software for the system, including games, small business applications, home finance, and other programs. The prices for the three models are $1495, $1695, and $1995, respectively.

Circle 274 on inquiry card.

CP/M Available for Pertec PCC 2000 Microcomputer

CP/M, a proprietary operating system developed by Digital Research Inc, will allow the PCC 2000 to run programs written in BASIC-80, FORTRAN-80, and COBOL-80. Basic facilities of CP/M include dynamic file management, a fast assembler, a general-purpose text editor, an advanced debugger, and several utilities. The price depends upon dealer configuration. Contact Pertec Computer Corp, 12910 Culver Blvd, Los Angeles CA 90066, for information.

Circle 276 on inquiry card.

New Products

Superchip Adds Text Processing Capabilities to Apple Computer

Superchip will enhance the text processing capabilities of the Apple computer, according to Eclectic Corp, 2830 Walnut Hill Ln, Dallas TX 75229. The read-only-memory device plugs into an Apple computer with no modification required and adds the full ASCII character set (including lowercase) plus thirty-one other non-ASCII characters. It also enhances editing capabilities for program and data modification. The device is compatible with Integer BASIC and Applesoft programs.

Used in conjunction with a special character edit cassette, also from Eclectic, Superchip allows the user to define new characters in a magnified format. With this feature, it is possible to create entire character sets, such as foreign alphabets, musical notation, and game pieces.

Superchip is $99.95 and the character edit cassette is $24.95.

Circle 277 on inquiry card.

Software Package on Health and Medical Information

VitaFacts is a series of health interest programs for microcomputer users. The first six programs are: Heart Attacks, Birth Control, Your Blood Pressure, Teenage Drinking and Drugs, Growing Up, and Talking About Sex. The program Teenage Drinking and Drugs includes information on all types and effects of drugs and liquor. Your Blood Pressure discusses the heart and its mechanics and functions, circulation, causes and effects of abnormal blood pressure, and more. The Birth Control program contains information on puberty, birth, and contraception.

Each package includes an audio cassette, a computer cassette, and a booklet. They have all been approved and endorsed by the College of Family Physicians of Canada. The price per package is $19.95. For more information, contact Richmond Software, 471 Richmond Rd, Ottawa Ontario, CANADA K2A 0G3.

Circle 278 on inquiry card.

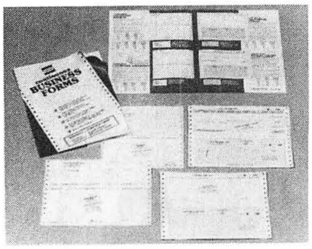
Tone Generator for TRS-80 Computer

The Telesis Tone/80 Programmable Tone Generator was designed for Radio Shack TRS-80 Level II personal computer owners. The Tone/80 responds to output commands from the TRS-80 and can produce 128 different tones. The unit can be used to create sound effects for games, compose musical tunes, or add sound to a burglar alarm.

The Tone/80 is assembled, tested and guaranteed, and requires no additional interfacing with the TRS-80. It includes a documented data and applications package with software for producing whistles, sirens, phasor sounds, and a few tunes. Also included are notes on how to add various sounds to your favorite computer games. The complete Tone/80 package is available for $89.95. Contact Telesis Laboratory, Peripherals Division, POB 1843, Chillicothe OH 45601.

Circle 279 on inquiry card.

New Products



Standardized Computer Forms

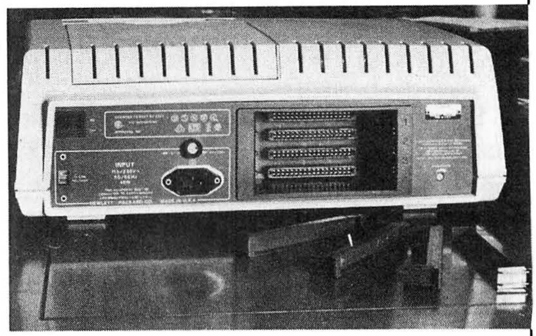
NEBS is offering a line of continuous computer checks, statements, and invoices. The forms are available with the name of the firm, address, and phone number in six quantity ranges from 500 thru 6000. Prices start at $14.95 for 500 statements; $32.50 for 500 two-part invoices; and $29.95 for 500 of either the payroll or all-purpose checks. At 6000-piece order levels, prices per thousand drop to $12.50, $33, and $22.50 respectively. They also offer custom personal checks for home computer systems users. For ordering information and free samples, write New England Business Service Inc, N Main St, Groton MA 01450.

Circle 281 on inquiry card.

Keyboard Expandor

This hardware/software modification transforms your Apple II microcomputer into a complete upper and lowercase system. Cap and Shift Locks are included; all Apple characters and monitor editing functions are maintained. Upper and lowercase can be used in Text files, in Print and Rem statements within BASIC programs, in file names, and in Immediate mode. The software runs in 250 bytes of memory. Cost is $20. Contact C&H Micro, POB 249, Clifton Park NY 12065.

Circle 282 on inquiry card.



Hewlett-Packard Introduces Personal Computer for Professionals

The new Hewlett-Packard HP-85 personal computer is here. It is designed for use in business and industry by engineers, scientists, accountants, and investment analysts. It can also be used in the home by hobbyists and as an instructional computer in secondary schools, colleges, and universities.

The system features a video terminal, printer, tape cartridge, and graphics capability in a package the size of a typewriter. It is equipped with four input/output (I/O) ports to allow the user to expand the system to include plotters, printers, disk drives, and other peripherals.

The HP-85 comes with 16 K bytes of programmable memory and can be expanded to 32 K by plugging an optional memory module into one of the ports on the back of the machine. The graphics display is useful to engineers for plotting functions and test analysis, and to businesspersons to plot statistics. The display on the screen can be easily printed out on the built-in printer. The system has a 5-inch high-resolution black and white video display with sixteen lines of display and 32 characters per line. The thermal printer, which operates in both alphanumeric and graphics modes, prints two 32- character lines per second. The HP-85 tape drive uses HP Data Cartridges, which have a capacity of 200 K bytes of memory each, and feature a tape directory that enables the system to automatically find exact tape locations of recorded programs and data.

The HP-85 is 41 by 46 by 15 cm (16 by 18 by 6 inches) and weighs under 20 pounds. It comes with a manual and a standard application software package which contains fifteen programs. The price of the HP-85 is $3250. For more information, contact Inquiries Manager, Hewlett-Packard Co, 1507 Page Mill Rd, Palo Alto CA 94304.

Circle 283 on inquiry card.

New Products

Space Shuttle Landing Simulator

This program is modeled after the NASA Shuttle Mission Simulator in Houston. It is a real-flight simulator (except for roll motion) with a visual display of the sky and ground. High-resolution graphics show the cockpit view using animation, projective geometry, and graphics to depict the runway, sky, ground, and distant mountains and clouds. The paddles control the pitch control and speed brakes. Speed, altitude, sink and climb rate, distance from the threshold, speed brake setting, glide slope, and angle of attack are also displayed. Warnings and messages are also displayed.

Functional features include angle of attack control, full stall capability, eject and eject warning, landing gear, speed brakes, and wheel brakes on rollout.

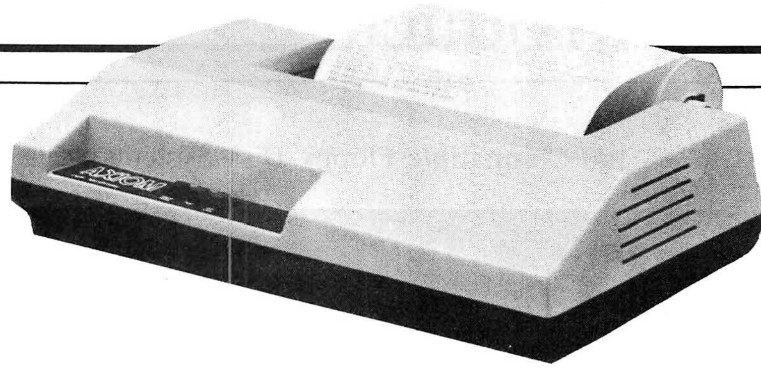
The program is available from Harvey's Space Ship Repair, POB 3478, University Park, Las Cruces NM 88003, for $15 on cassette. A floppy-disk version is also available.

Circle 284 on inquiry card.

Information on TRS-80

Guide to TRS-80 Information is a concise publication written for Radio Shack TRS-80 owners and users. The guide covers: address, articles, authors, bugs, comments, hardware, ideas, phone numbers, predictions, reduced prices, rumors, software sources, tips, used units and user groups. Price is $3. Contact F E Huebner, POB 37206, Oak Park MI 48237.

Circle 285 on inquiry card.



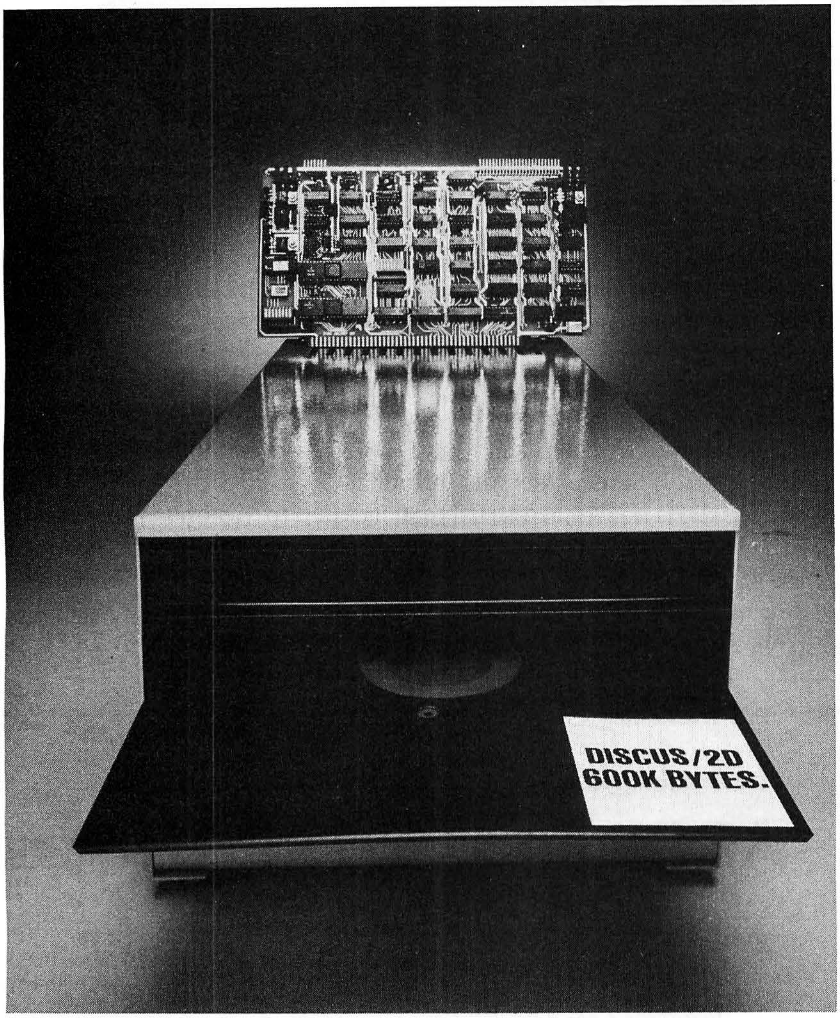
Low Profile Impact Printers Announced by Axiom

The IMP series of printers can print 80, 96, or 132 columns at one line per second. The IMP-1, which has a friction feed mechanism, can make three copies on 21.5 cm (8.5 inch) wide paper for word processing, and also handles Teletype rolls. IMP-2 has friction and tractor feed. Both units have 7 by 7 dot matrix, 96-character ASCII set, double-width characters, and the IMP-2 can do graphics under software control. Standard inputs include a Centronics parallel interface and RS-232C and 20 mA serial interfaces up to 1200 bits per second; 512-character buffer, with 2 K bytes of characters optional. The units are 45 by 22 by 9 cm (17 by 8.75 by 3.5 inches) and weigh 15 pounds apiece. The IMP-1 costs $695; the IMP-2 is $795. They are available from Axiom Corp, 5932 San Fernando Rd, Glendale CA 91202.

Circle 286 on inquiry card.

New Products

S-100 Compatible Floppy-Disk Systems



The DISCUS 2+2 is a quad-density floppy-disk system capable of storing up to 1.2 M bytes using the IBM compatible soft-sectored format and double-sided floppy-disk drives. The DISCUS 2D is a double-density floppy-disk system with a capacity of 600 K bytes also using the soft-sectoring IBM format and a single-sided, 8-inch floppy-disk drive. The DISCUS 1 is a single-density system with a capacity of 250 K bytes of storage using the IBM soft-sectored format. The Disk Jockey 2D S-100 compatible controller is used with all three systems and includes a serial input/output (I/O) port, 1 K bytes of programmable memory, and 1 K bytes of read-only memory which is programmed to execute functions such as read sector, write sector, bootstrap load, and more. The systems come with power supplies, line cords, signal cables, and a disk operating system called Disk/ATE, and a BASIC interpreter called BASIC/V which has virtual memory addressing capabilities. The price is $595 for the DISCUS 1, $1149 for the DISCUS 2D, and $1545 for the DISCUS 2+2. Contact Thinker Toys, 5221 Central Ave, Richmond CA 94804, for more information.

Circle 287 on inquiry card.

A Home Computer or Computerized Home?

A new system figures taxes, plays games, recalls recipes, and turns a new home into a computer with a completely programmable electrical system. It will turn lights on and off by program, control appliances, fire and security systems, and remind you of anniversaries or other appointments.

The Harris Labs Home Control System is a low-power system that can be installed in any housing start by an electrical contractor. It can control up to 256 circuits with expansion capability built in. A TMS9900 16-bit microprocessor runs the system with memory units containing operating systems supporting application programs in a multitask environment.

The price ranges from $3000 to $6000, depending on the size of the system. For more information, contact Harris Labs, (515) 753-8529 or Texas Instruments Inc, (214) 238-4783.

Circle 288 on inquiry card.

A 12-Slot S-100 Mainframe for System Builders

This S-100 mainframe box measures 29 cm high by 17 cm wide by 46 cm deep (11.6 by 7 by 18 inches) and comes with a fan and a circuit breaker. The power supply supports output voltages of +8 V DC at 20 A and ± 16 V DC at 4 A. Input may be 105, 115, or 125 V AC. The product is priced at $399.95, from California Computer Systems, 309 Laurelwood Dr, Santa Clara CA 95050.

Circle 289 on inquiry card.

New Products

Automatic Printing of Disk Programs in Paged Format

Data Associates, POB 882, Framingham MA 01701, has announced PAGER8, a program that provides program listings in a paged format ideal for manuals, publications, and for convenient editing. It is written for the TRS-80 and runs on a single floppy-disk system with 32 K bytes of memory. PAGER8 reads any named BASIC program saved in ASCII form on disk and prints it on the printer. The page length, lines per page, characters per line, and left border size can all be changed within the program through prompting instructions. The program is self-documenting with built-in checks for entry errors. PAGER8 comes with a manual and three copies on cassette for $19.95.

Circle 290 on inquiry card.

Cassette-Based Word Processor

Designed for TRS-80 Level II microcomputers and compatible with the Radio Shack Line Printer II, a new word processing program also runs on the Centronics Model 730 and other 730 series Centronics printers. The program has the ability to enter text from keyboard or cassette, to display text starting with any line, to change or edit any selected line, to scroll text forward or backward, to insert or delete any line, to add to end of text, to save text on tape, and to line-print text. A number of options are selectable by the user. The program is available for $19.95 from GB Associates, POB 3322, Granada Hills CA 91344.

Circle 291 on inquiry card.

Coming Events

March, Short Courses on Microcomputers, The George Washington University, Washington DC. Contact the Director of Continuing Engineering Education, George Washington University, Washington DC 20052 or call (202) 676-6106 or (800) 424-9773.

March 1, Exploring Small Computers, Albion College, Albion MI. This fair will feature exhibits and seminars on microcomputers and their applications in business, education, and the home. Contact D W Kammer, Dept of Physics, Albion College, Albion MI 49224.

March 3-5, Office Automation Conference, Georgia World Congress Center, Atlanta GA. A combination conference and exhibition of office computer systems has been developed to help management understand the growing technology of business computer systems. For more information, contact H A Bruno and Associates Inc, 78 E 56th St, New York NY 10022.

March 10-12, 1980 National Office Exhibition and Conference, Automotive Building on Exhibition Place, Toronto CANADA. Subject areas of the conference will include energy conservation, small business computers, micrographics, word processing, telecommunications, copiers, office landscaping, and many others. There will be around one hundred exhibitors presenting their products and giving demonstrations. For more information, contact Whitsed Publishing Ltd, Suite 2504, 2 Bloor St W, Toronto Ontario CANADA M4W 3E2.

March 15, Annual PACS Computer Games Festival, LaSalle College Ballroom, 20th and Olney, Philadelphia PA. This event is sponsored by the Philadelphia Area Computer Society and the LaSalle College Physics Dept. It goes from 10 AM to 6 PM. For further information, contact Stephen A Longo, Physics Dept, LaSalle College, Philadelphia PA 19141, (215) 951-1255.

March 24-28, Fourth European Conference on Electrotechnics, Stuttgart. This conference will review recent developments, trends, and applications in the field of microelectronics. Microprocessors, computer communication, industrial electronics, applications of microelectronics in the automobile and in medicine, and other topics will be covered. The conference language will be English. Contact Prof Dr W E Proebster, IBM Deutschland GmbH, Postfach 80 08 80, D-7000 Stuttgart 80 WEST GERMANY (BRD).

April 30-May 2, Computerized Office Equipment Expo, O'Hare Exposition Center, Rosemont IL. The latest developments in computers, word processors, copiers/duplicators, telephone systems, and other business equipment will be featured. The seminars will cover guidelines on buying computer systems, telephone and copier systems, the use of word processors, and more. Contact Industrial and Scientific Conference Management Inc, 222 W Adams St, Chicago IL 60606.

coming in onComputing

Word Processing — one of the hottest topics in personal computing. onComputing evaluates 16 different word processors and text editors.

The Nestar System — A new approach to computers in the classroom lets different brands of personal computers talk to one central source of programs and data.

What Is a Floppy Disk?

APF's New Personal Computer: The Imagination Machine

And much more, including the regular onComputing features.

InterAction Results

The winning article in the Fall 1979 issue of onComputing was "A Printer Primer" by Elizabeth Hughes. Second place went to Steve Ciarcia's review of "The Centronics Printer". Prizes of $100 and $50 will be awarded to the respective authors. In third place was "Let Us Assemble" by Len Lazar. If you would like to vote for your favorite articles in this issue, simply fill out the card on the opposite page.

interAction

Reader Service

To get further information on the products advertised in onComputing, fill out the reader service card with your name and address. Circle the appropriate numbers for the advertisers you select from the list, and drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by onComputing.

READER SERVICE

Winter 1980

For fastest service transfer mailer label from wrapper to coupon provided below. Requests cannot be honored unless zip code is given. This card valid for 90 days only.

NOTE—If label is missing or defaced fill out coupon carefully—PLEASE PRINT—this is the only way to get requested material to you.

Name ______________________________

(Title) ______________________________

(Company) ______________________________

Address ______________________________

City ______________ State __________ Zip Code __________

1	21	41	61	81	101	121	141	161	181	201	221	241	261	281
2	22	42	62	82	102	122	142	162	182	202	222	242	262	282
3	23	43	63	83	103	123	143	163	183	203	223	243	263	283
4	24	44	64	84	104	124	144	164	184	204	224	244	264	284
5	25	45	65	85	105	125	145	165	185	205	225	245	265	285
6	26	46	66	86	106	126	146	166	186	206	226	246	266	286
7	27	47	67	87	107	127	147	167	187	207	227	247	267	287
8	28	48	68	88	108	128	148	168	188	208	228	248	268	288
9	29	49	69	89	109	129	149	169	189	209	229	249	269	289
10	30	50	70	90	110	130	150	170	190	210	230	250	270	290
11	31	51	71	91	111	131	151	171	191	211	231	251	271	291
12	32	52	72	92	112	132	152	172	192	212	232	252	272	292
13	33	53	73	93	113	133	153	173	193	213	233	253	273	293
14	34	54	74	94	114	134	154	174	194	214	234	254	274	294
15	35	55	75	95	115	135	155	175	195	215	235	255	275	295
16	36	56	76	96	116	136	156	176	196	216	236	256	276	286
17	37	57	77	97	117	137	157	177	197	217	237	257	277	297
18	38	58	78	98	118	138	158	178	198	218	238	258	278	298
19	39	59	79	99	119	139	159	179	199	219	239	259	279	299
20	40	60	80	100	120	140	160	180	200	220	240	260	280	300

interAction

Winter 1980

interAction is your direct line to the editor's desk. Each month, the two top rated authors receive bonuses based on your votes. To use this card, refer to the list of authors, titles, and corresponding article numbers on the opposite page. Then rate each article on a scale from 0 to 10 below by circling the appropriate rating number to the right of each article number. Your feedback helps us produce the best possible magazine each month.

.... Rating ....

ARTICLE NUMBER		POOR		FAIR	GOOD		VERY GOOD			EXCELLENT	
1	0	1	2	3	4	5	6	7	8	9	10
2	0	1	2	3	4	5	6	7	8	9	10
3	0	1	2	3	4	5	6	7	8	9	10
4	0	1	2	3	4	5	6	7	8	9	10
5	0	1	2	3	4	5	6	7	8	9	10
6	0	1	2	3	4	5	6	7	8	9	10
7	0	1	2	3	4	5	6	7	8	9	10
8	0	1	2	3	4	5	6	7	8	9	10
9	0	1	2	3	4	5	6	7	8	9	10
10	0	1	2	3	4	5	6	7	8	9	10
11	0	1	2	3	4	5	6	7	8	9	10
12	0	1	2	3	4	5	6	7	8	9	10
13	0	1	2	3	4	5	6	7	8	9	10
14	0	1	2	3	4	5	6	7	8	9	10
15	0	1	2	3	4	5	6	7	8	9	10
16	0	1	2	3	4	5	6	7	8	9	10
17	0	1	2	3	4	5	6	7	8	9	10
18	0	1	2	3	4	5	6	7	8	9	10
19	0	1	2	3	4	5	6	7	8	9	10
20	0	1	2	3	4	5	6	7	8	9	10

Comments ______________________________

____

____

____

onComputing™

Place
Stamp
Here

onComputing
70 Main St
Peterborough NH 03458
USA

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS Permit Number 7976 New York NY 10001
POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
POB 1911 GPO
New York NY 10001
USA

onComputing™

The Personal Computer Line by OHIO SCIENTIFIC

Personal Computers

C1P: $349 A dramatic breakthrough in price and performance. Features OSI's ultra-fast BASIC-in-ROM, full graphics display capability, and large library of software on cassette and disk, including entertainment programs, personal finance, small business, and home applications. It's a complete programmable computer system ready to go. Just plug-in a video monitor or TV through an RF converter, and be up and running. 15K total memory including 8K BASIC and 4K RAM—expandable to 8K.

C1P MF: $995 First floppy disk based computer for under $1000! Same great features as the C1P plus more memory and instant program and data retrieval. Can be expanded to 32K static RAM and a second mini-floppy. It also supports a printer, modem, real time clock, and AC remote interface, as well as OS-65D V3.0 development disk operating system.

Professional Portables

C4P: $698 The professional portable that has over three times the display capability of C1Ps. Features 32 x 64 character display in up to 16 *colors,* graphics, audio output, a DAC for voice and music generation, key pad and joystick interfaces, AC remote control interface and much more. Utilizes a 4-slot BUS (2 used in base machine), 8K BASIC-in-ROM, 8K of *static* RAM and audio cassette interface. Can be directly expanded to 32K static RAM and two mini-floppy disks.

C4P MF: $1695 The ultimate portable computer has all the features of the C4P plus real time clock, home security system interface, modem interface, printer interface, 16 parallel lines and an accessory BUS. The standard machine operates at twice the speed of currently available personal computers (with GT option it runs even faster!). The C4P MF starts with 24K RAM and a single mini-floppy and can be directly expanded to 48K and two mini-floppies. Available software includes games, personal, business, educational and home control applications programs as well as a real time operating system, word processor and a data base management system.

Computers come with keyboards and floppies where specified. Other equipment shown is optional.

Home/Small Business Systems

C8P: $895 Same great features as the C4P in a tremendously expandable "mainframe package." Features over three times the expansion capability of the C4P for advanced home and demanding business applications. Can be expanded to 48K RAM, dual 8″ floppies, hard (Winchester) disks and multiple I/O devices such as Voice I/O and a universal telephone interface.

C8P DF: From $2597 The ultimate Home/Very Small Business Computer at a personal computer price. Features 32K RAM (expandable to 48K) and dual 8″ floppy disks (stores eight times as much information as a mini-floppy). Has all personal computer capabilities including 32 x 64 display, color graphics, sound, DAC, joystick interfaces, home features including real time clock, AC remote interface, home security and fire detection interface and can be expanded to include voice I/O and a universal telephone system for answering and initiating calls! Its large memory capability and 8″ floppies allow it to run most Ohio Scientific business system software including a compete accounting system, word processor and information management system.

For literature and the name of your local dealer, CALL 1-800-321-6850 TOLL FREE.

OHIO SCIENTIFIC
1333 SOUTH CHILLICOTHE ROAD
AURORA, OH 44202 • [216] 831-5600

Circle 280 on inquiry card.